SAPIENZA
UNIVERSITÀ DI ROMA

Protocols for EH-WSNs

Internet of Things, a.a. 2018/2019

Un. of Rome "La Sapienza"

Chiara Petrioli[†]

[†] *Department of Computer Science – University of Rome "Sapienza" – Italy*

SAPIENZA
UNIVERSITÀ DI ROMA

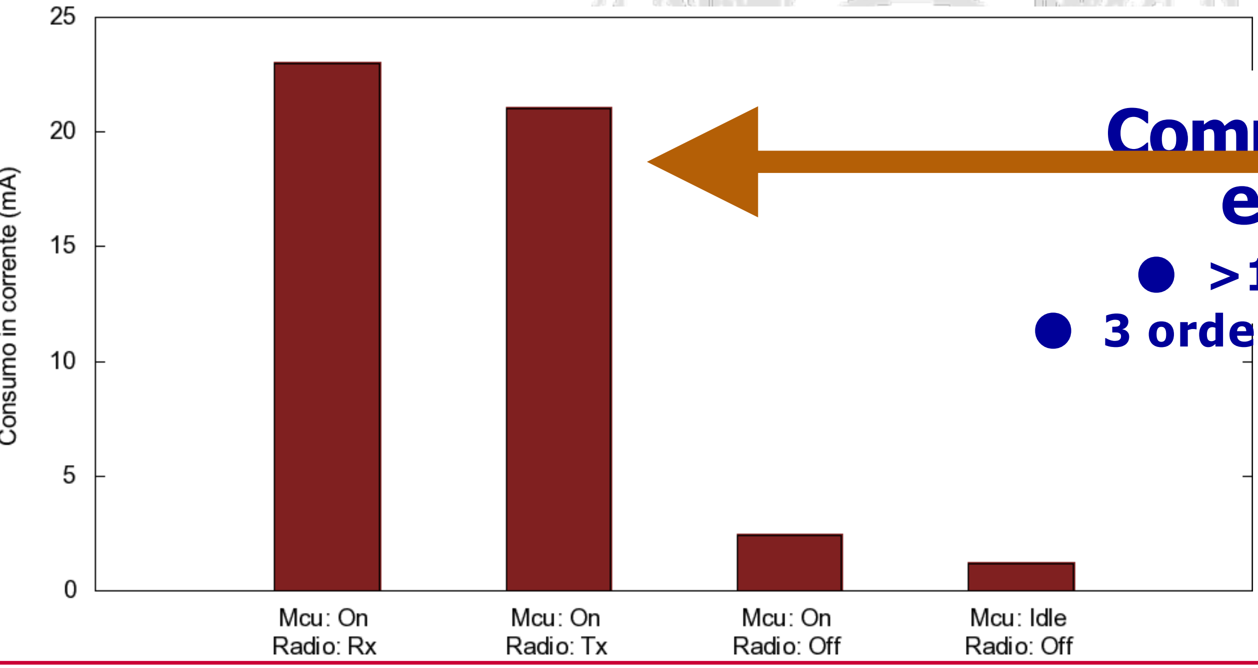
Application scenarios

Structural health monitoring

Cultural Heritage

Energy Consumption

- In many applications (e.g., SHM) the network is required to run for **decades**
- Nodes are powered by batteries
  - Limited lifetime (a few days on 2xAA batteries if always on)

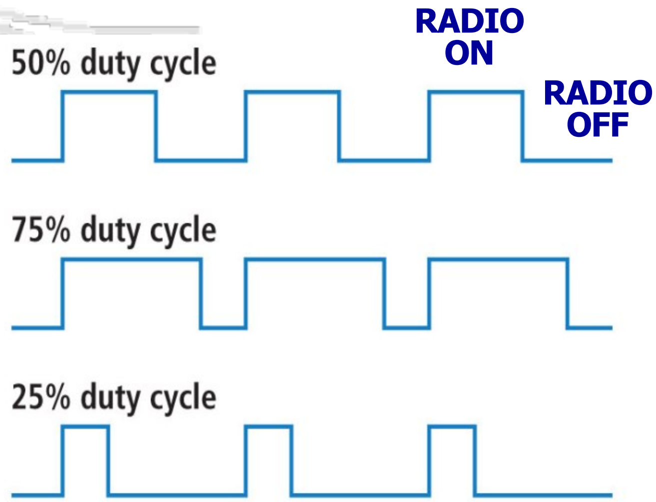


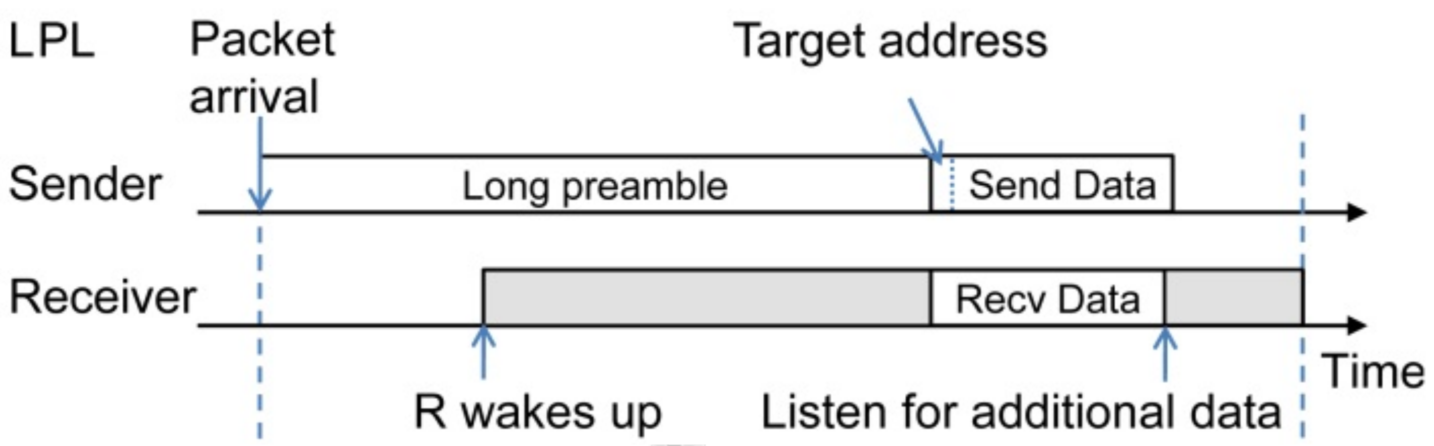
Communication is expensive!

- **>10x w.r.t. MCU on**
- **3 orders of magnitude w.r.t. sleep**

Standard Approach: Duty Cycling

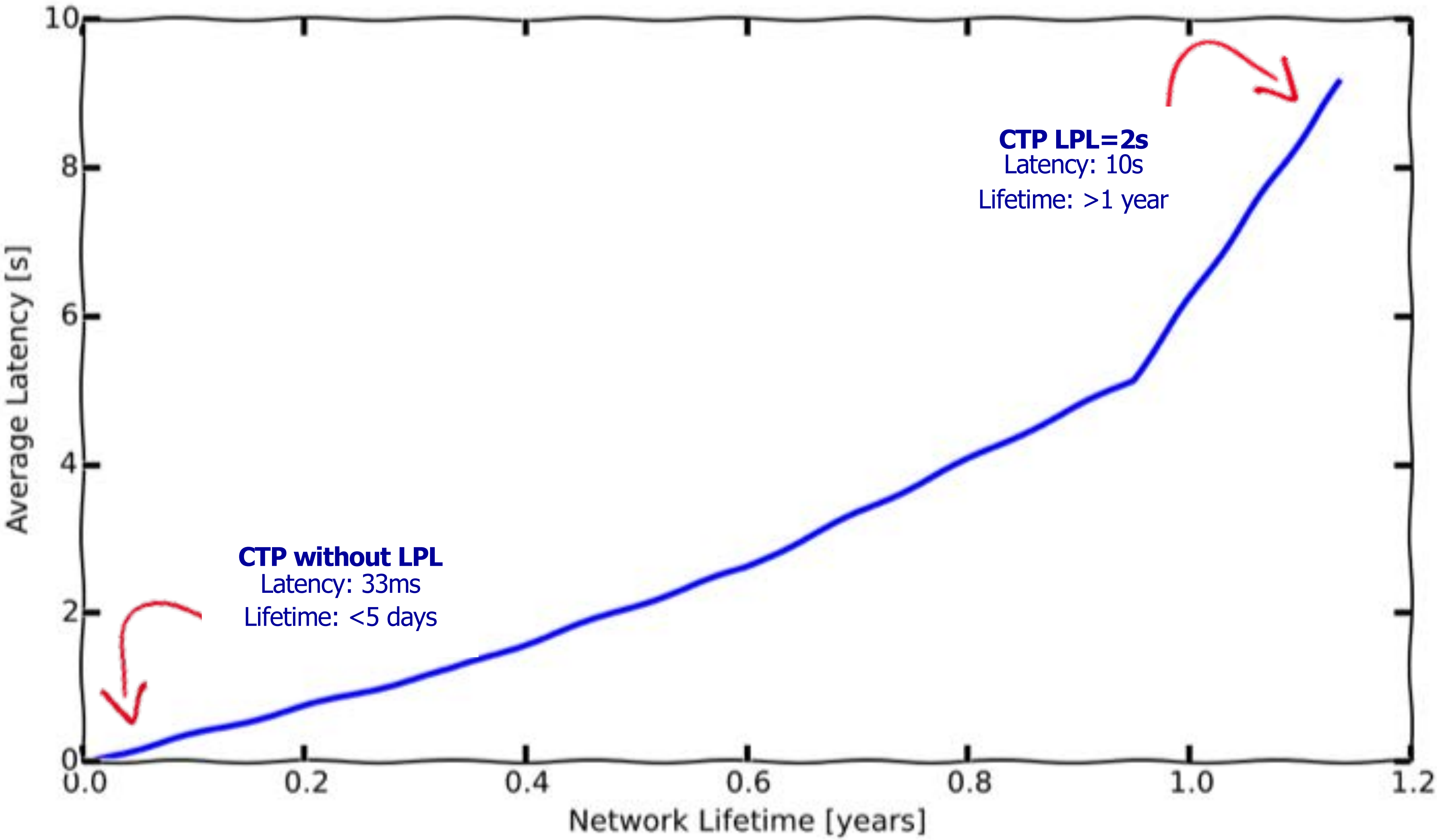
- Periodically cycle the radio between ON/OFF states
  - OFF = save energy, but no communication
  - ON = high energy, but data can be transmitted and received



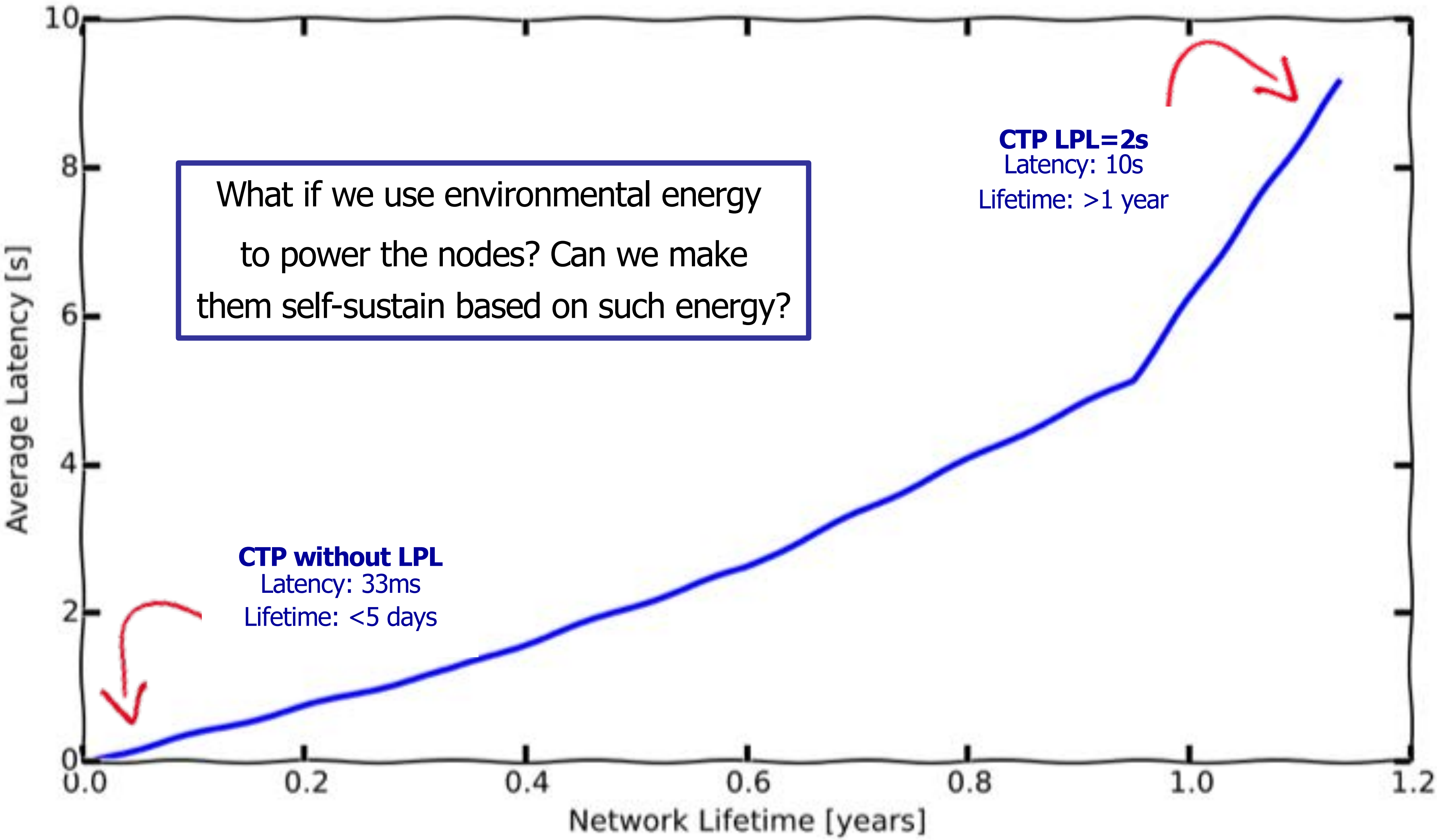


SAPIENZA
UNIVERSITÀ DI ROMA

Latency vs. Energy Trade-off

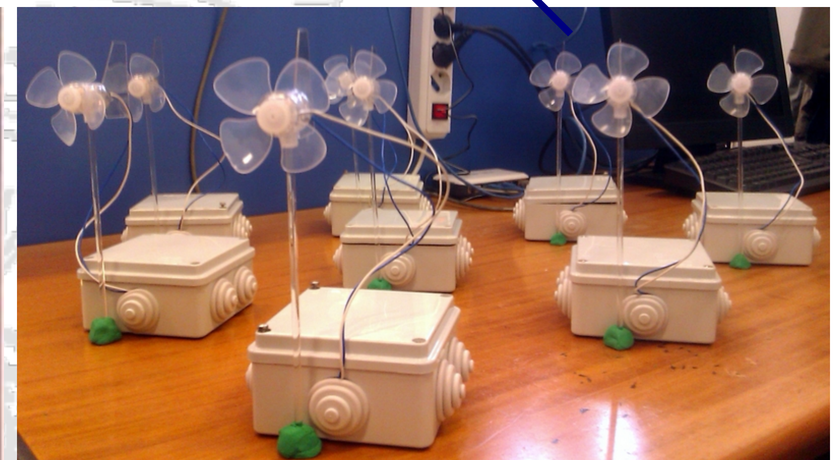


Latency vs. Energy Trade-off



What if we use environmental energy to power the nodes? Can we make them self-sustain based on such energy?

EH-WNS



- Pose the basis for very long lasting operation
- Energy Neutral protocols have been proposed for several applications
- Changes also what a WSN can do

EH-WNS



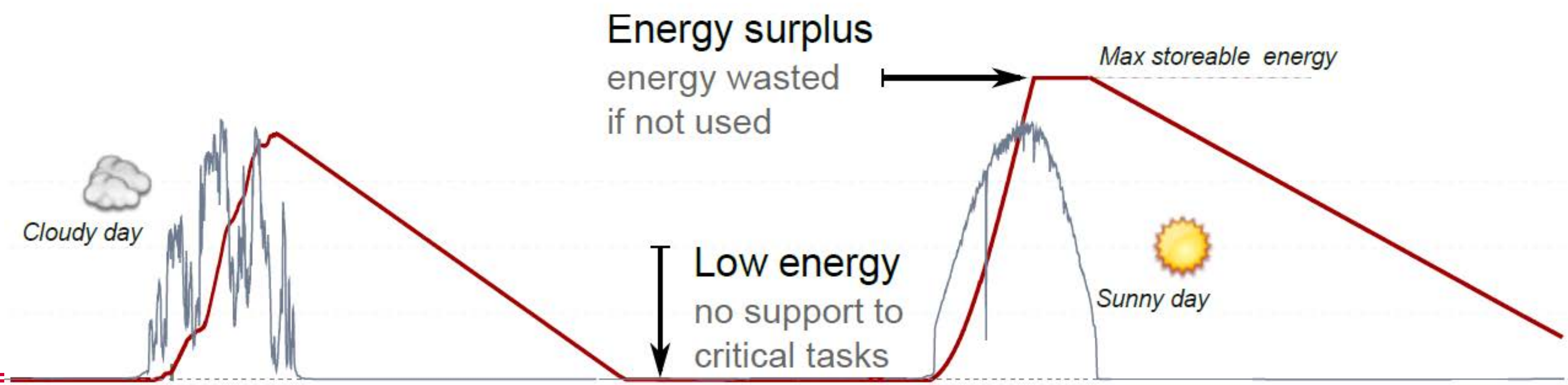

- Pose the basis for very long lasting operation
- Energy Neutral protocols have been proposed for several applications
- Changes also what a WSN can do

Pro-Energy

Alessandro Cammarano, Chiara Petrioli, and Dora Spenza, **Online Energy Harvesting Prediction in Environmentally Powered Wireless Sensor Networks**. In ***IEEE Sensors Journal***, Volume 16, Issue 17, pp. 6793 - 6804, Sep 2016.

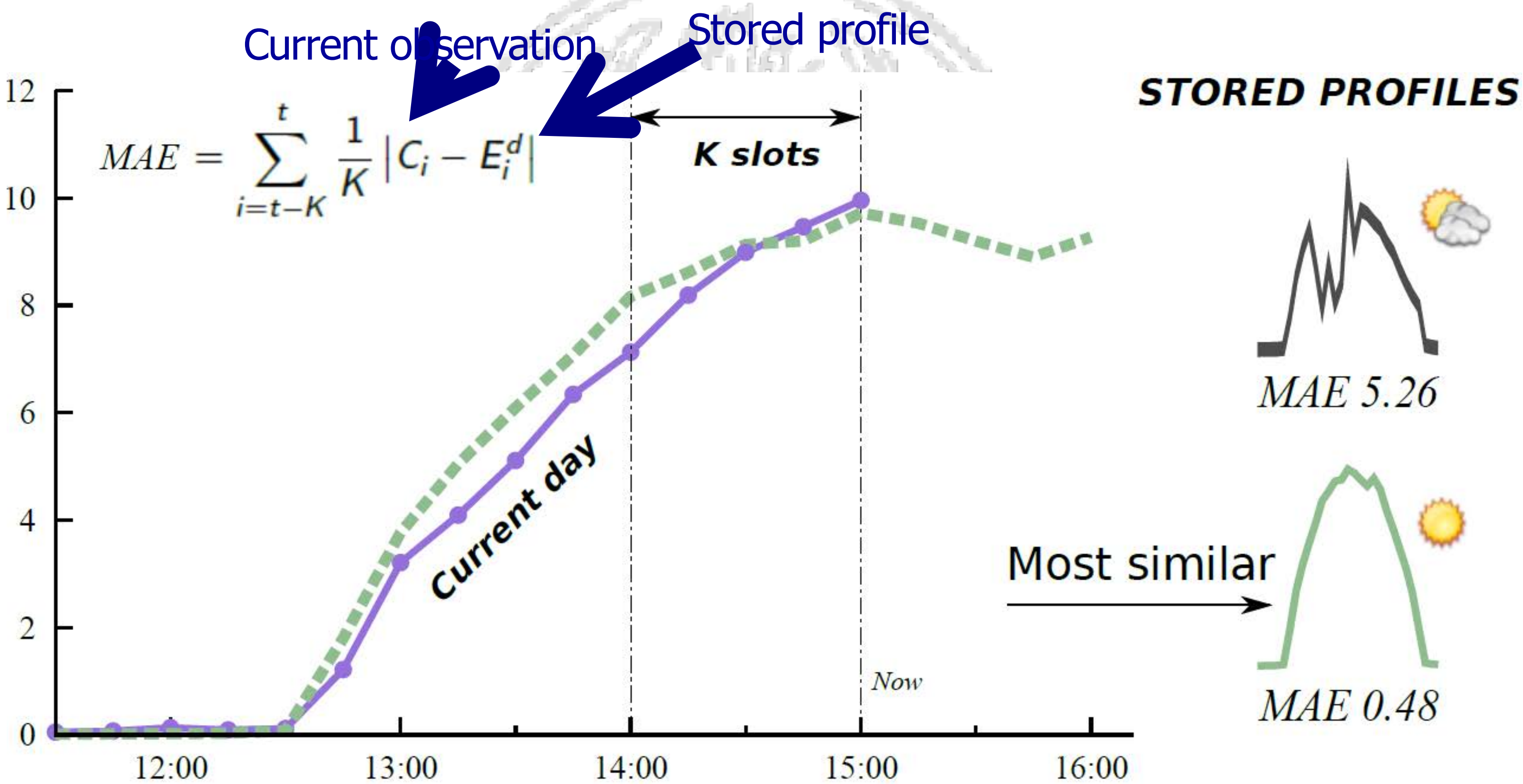
Why energy predictions?

- Energy predictions to mitigate uncertain energy availability
- Plan energy usage in advance: **proactive** vs reactive energy allocation
- Exploit available energy at best:
  I. Minimizing the likelihood of running out of energy and missing high priority tasks
  II. Minimizing the waste of energy (energy buffers are limited in size and time)
  III. Enable operations which were not considered feasible



Pro-Energy in a nutshell

- Keep track of energy profiles observed during D typical days
- Store traces representative of different weather conditions (sunny, windy, …)
- Predict future energy intake by looking at the most similar stored profile



Pro-Energy in a nutshell

$$\hat{E}_{t+1} = \alpha \cdot C_t \; + \; (1-\alpha) \cdot E^d_{t+1} \qquad (2)$$

where:

- $\hat{E}_{t+1}$ is the predicted energy intake in timeslot $t+1$ of the current day;
- E^d_{t+1} is the energy harvested during timeslot $t+1$ on the stored day d;
- C_t is the energy harvested during timeslot t on the current day C;
- α is a weighting factor, $0 \leq \alpha \leq 1$.

The weighting parameter, α, allows to combine the value reported in the stored profile with the current energy observation, i.e., the energy observed in the last slot, C_t.

Medium Term Energy Predictions

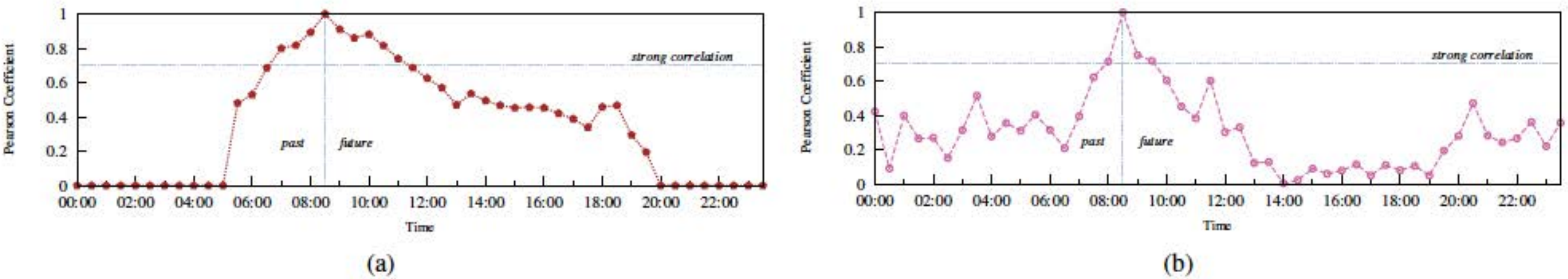


Figure 2. Pearson autocorrelation coefficient for (a) solar ORNL Dataset and (b) wind Bologna Dataset.

$$\gamma_i = \begin{cases} \alpha \cdot \left(1 - \frac{i-1}{G}\right), & \text{if } i \leq G \\ 0 & \text{if } i > G \end{cases} \qquad \forall i, 1 \leq i \leq F$$

where:

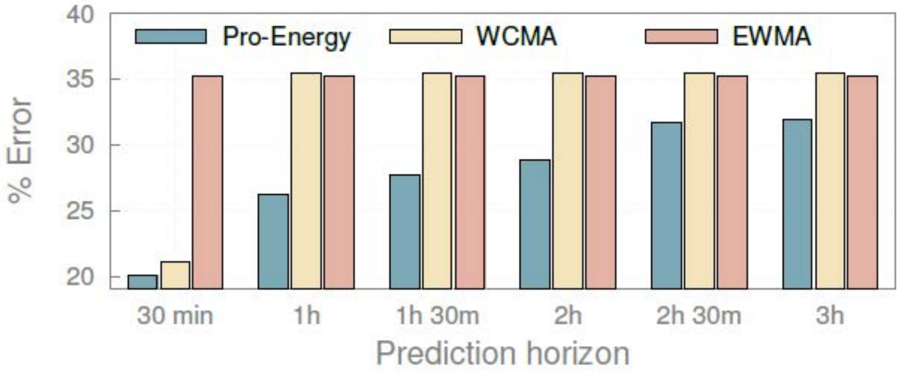
- α is the weighting factor defined in Equation (2);
- i is the i^{th} timeslot in the future, with respect to the current slot, t;
- G is the number of timeslots in the future which show a correlation above a given threshold with timeslot t;
- F is the number of future timeslots for which Pro-Energy is delivering energy predictions.

Medium term energy prediction estimation

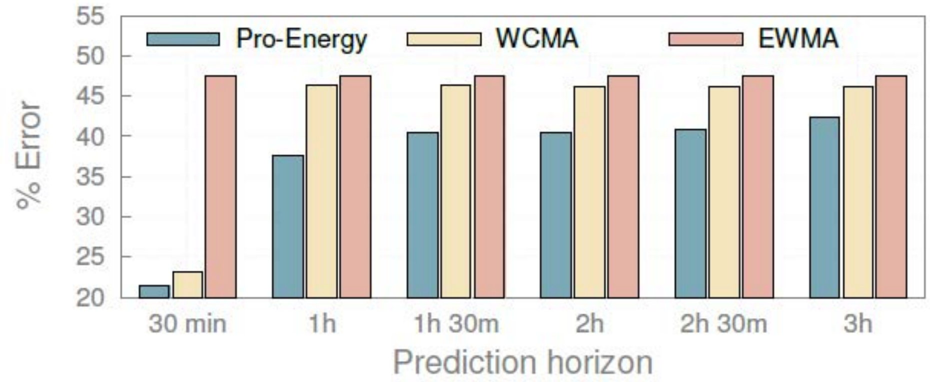
$$\hat{E}_{t+i} = \gamma_i \cdot C_t \; + \; (1 - \gamma_i) \cdot E^d_{t+i}$$

Pro-Energy accuracy

Solar



Wind



Solar: Pro-Energy performs up to **75%** better than EWMA and **60%** better than WCMA

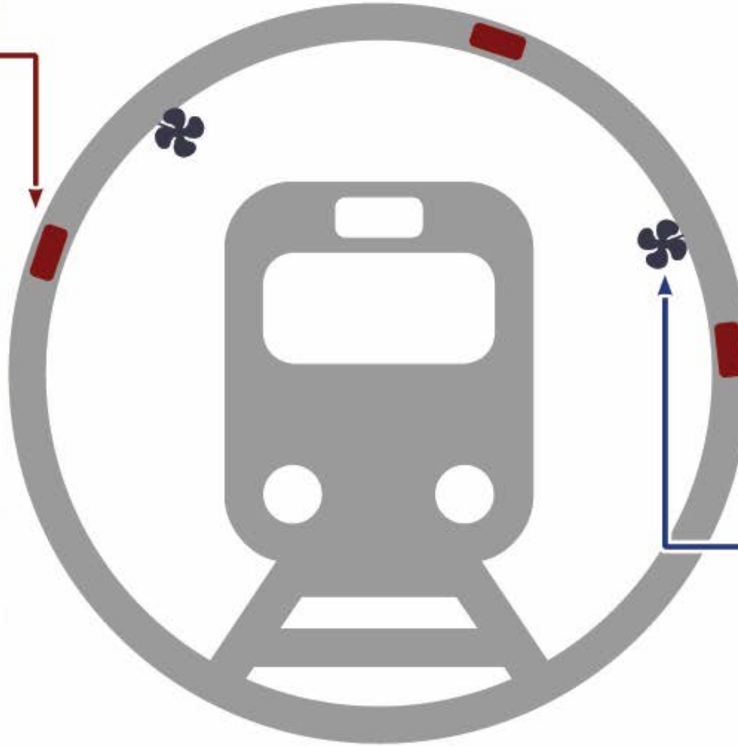
Wind: Pro-Energy performs up to **55%** better than EWMA and **10%** better than WCMA

Rome underground testbed

Vibrating Wire Strain Gauges

Monitor concrete and steel deformations to evaluate stability of the underground tunnel

SISGEO OVK4200VC00
Dedicated interface board for TelosB mote
Energy consumption: 720 mJ



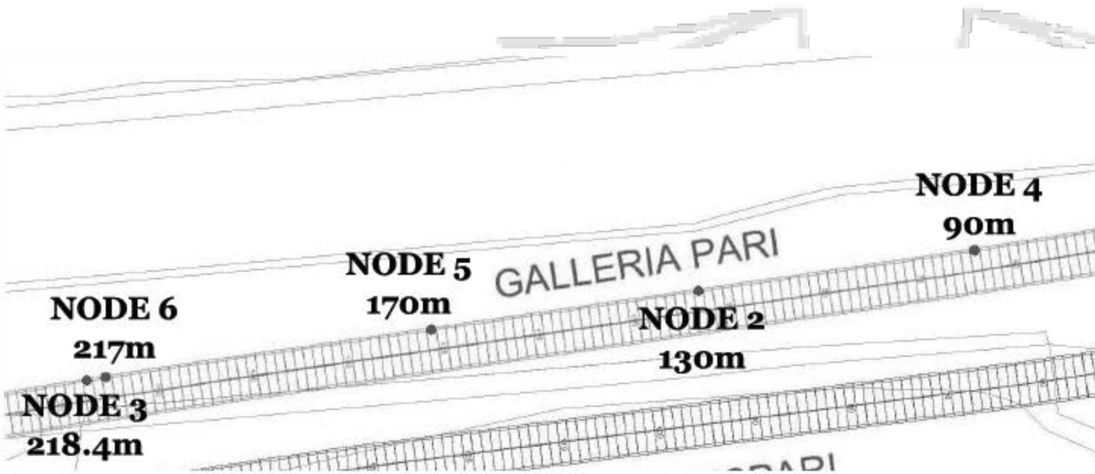
Micro wind turbines

Energy harvesting from wind generated by trains

Why air-flow energy harvesting?

- SHM sensors are power-hungry
- required lifetime of decades or more
- battery-powered WSNs last only a few years

In-field air-flow data collection



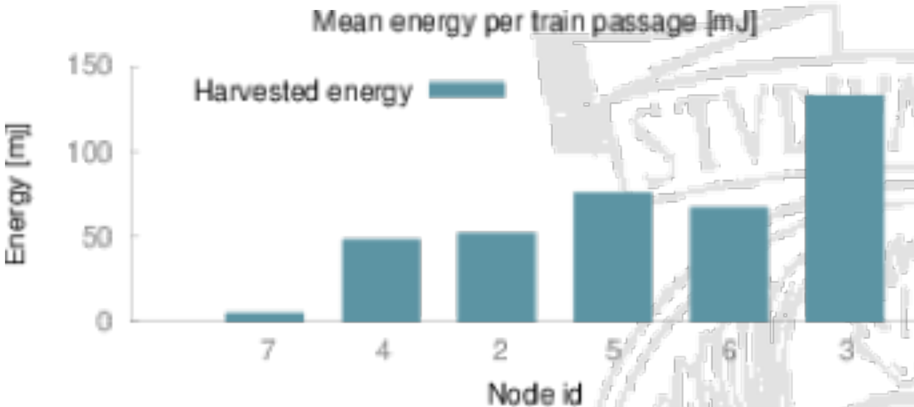
Site plan courtesy of Roma Metropolitane and Tre Esse Engineering (FP7 project GENESI).

220 meter of instrumented tunnel

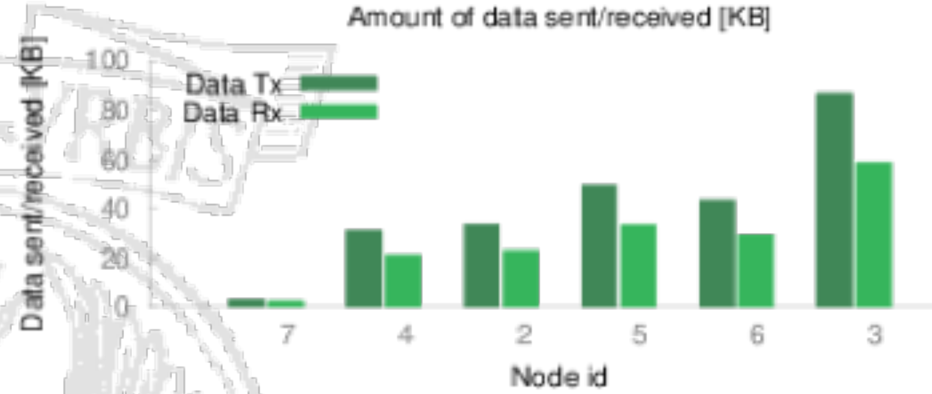
6 energy-harvesting nodes

33 days of data collection

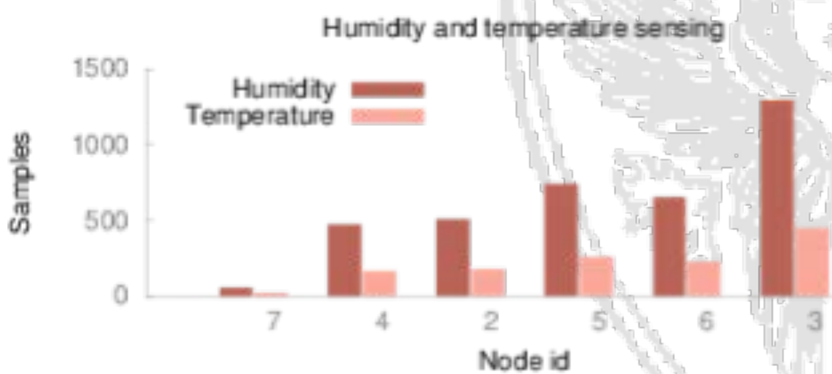
Feasibility study



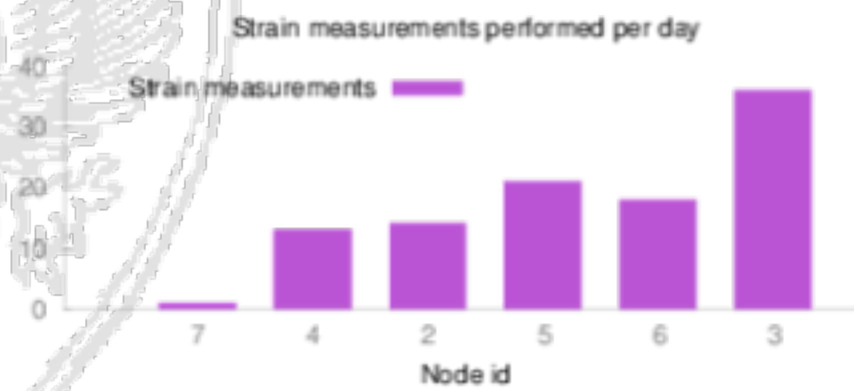
Up to 133 mJ harvested per train passage



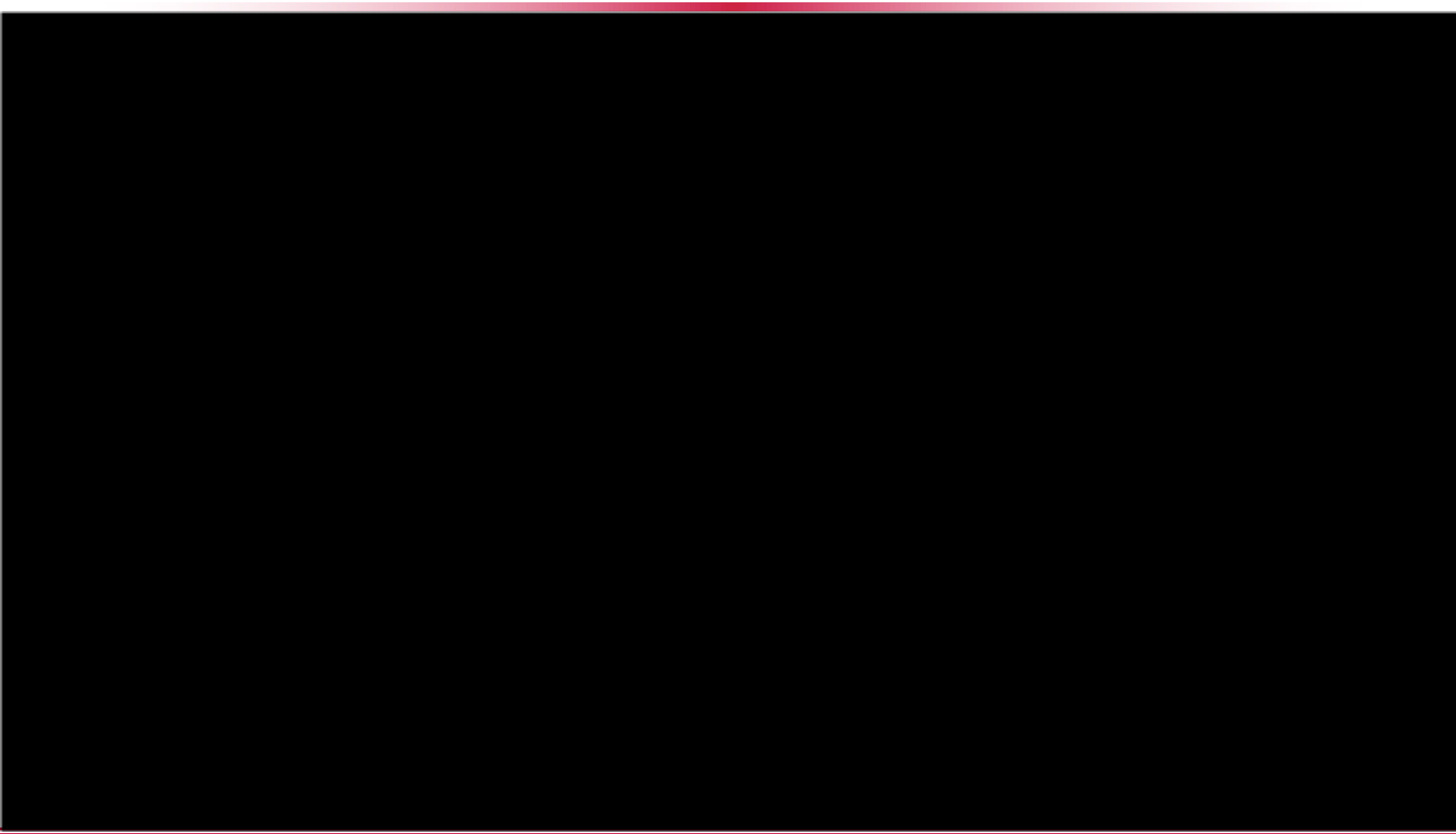
Transmit/receive tens of KB



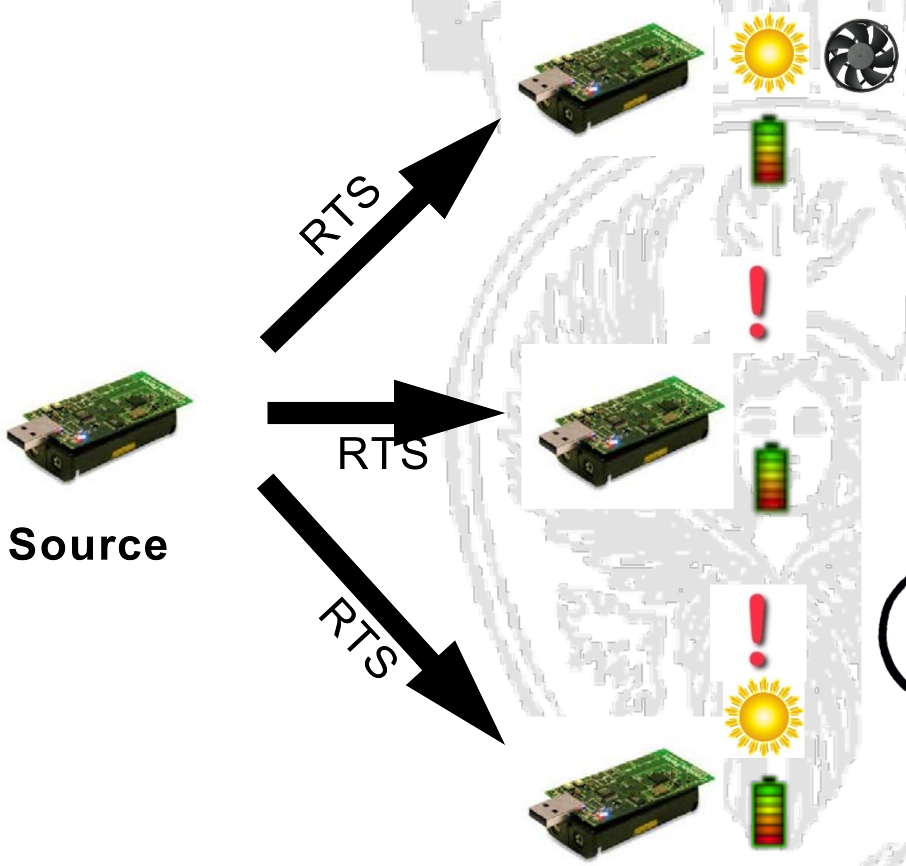
Collect hundreds of humidity and temperature samples



Up to 36 strain measurements per day



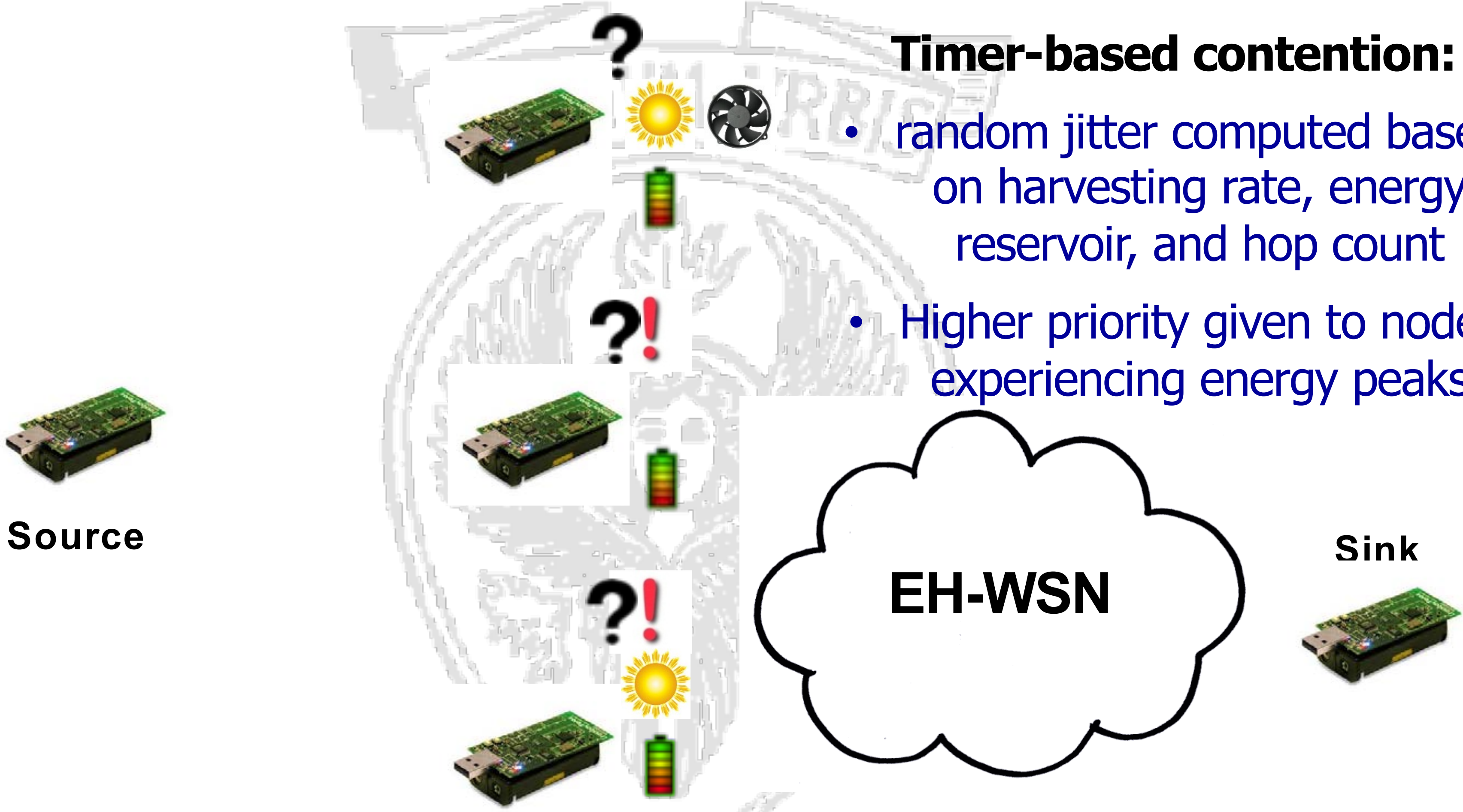
Harvesting-aware routing



Timer-based contention:

- random jitter computed based on harvesting rate, energy reservoir, and hop count
- Higher priority given to nodes experiencing energy peaks

Harvesting-aware routing



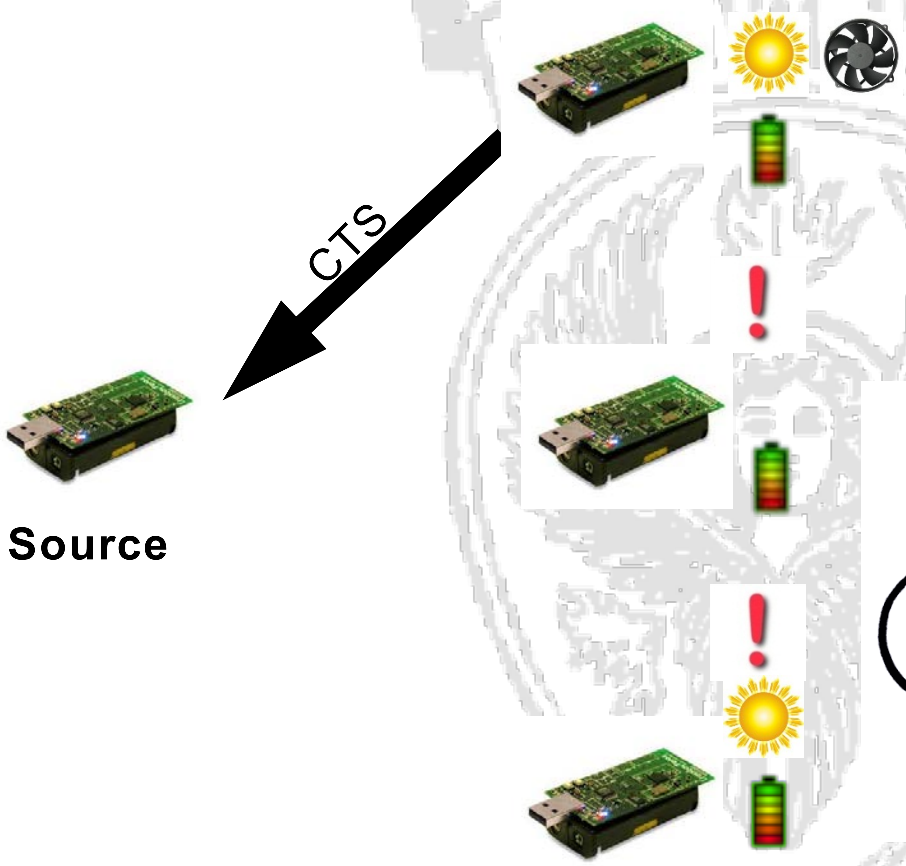
Timer-based contention:

- random jitter computed based on harvesting rate, energy reservoir, and hop count
- Higher priority given to nodes experiencing energy peaks

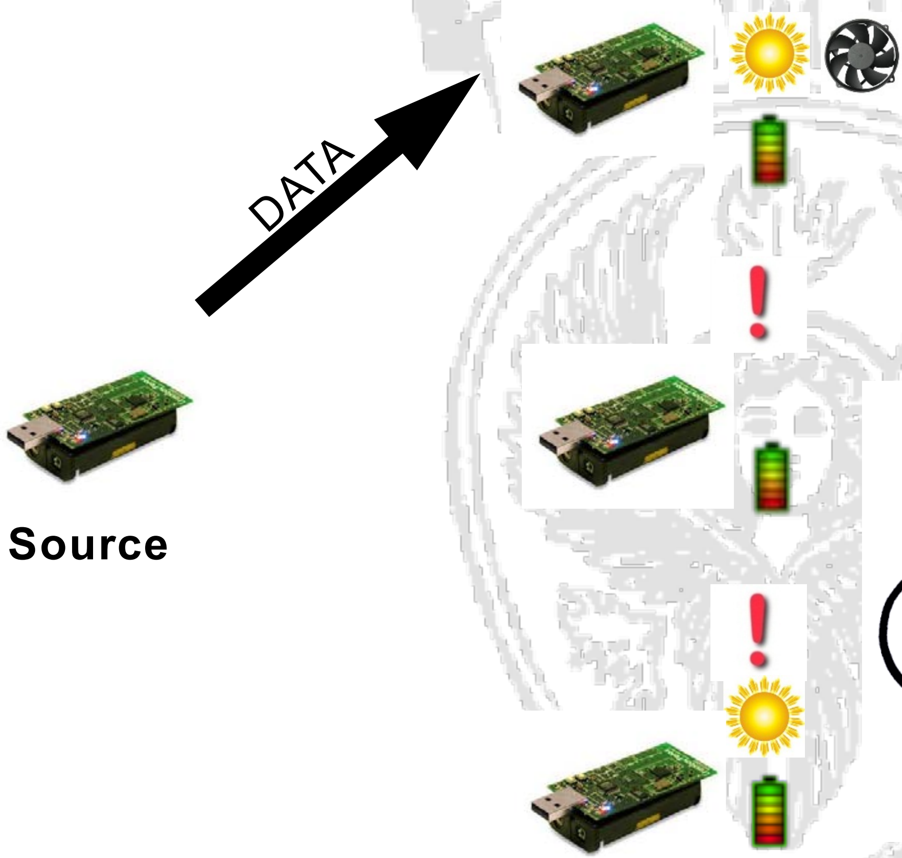
Harvesting-aware routing

Timer-based contention:

- random jitter computed based on harvesting rate, energy reservoir, and hop count
- Higher priority given to nodes experiencing energy peaks



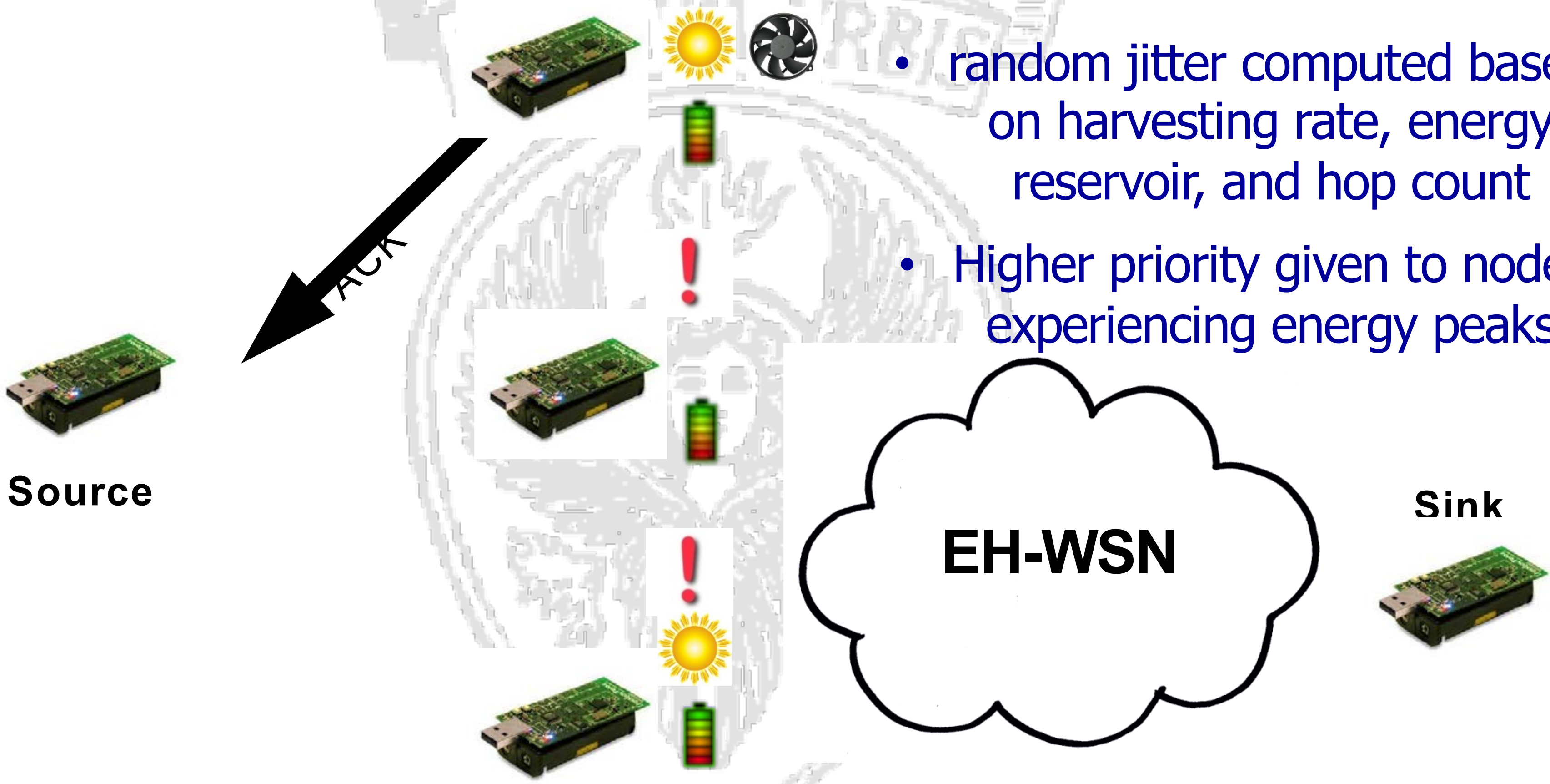
Harvesting-aware routing



Timer-based contention:

- random jitter computed based on harvesting rate, energy reservoir, and hop count
- Higher priority given to nodes experiencing energy peaks

Harvesting-aware routing

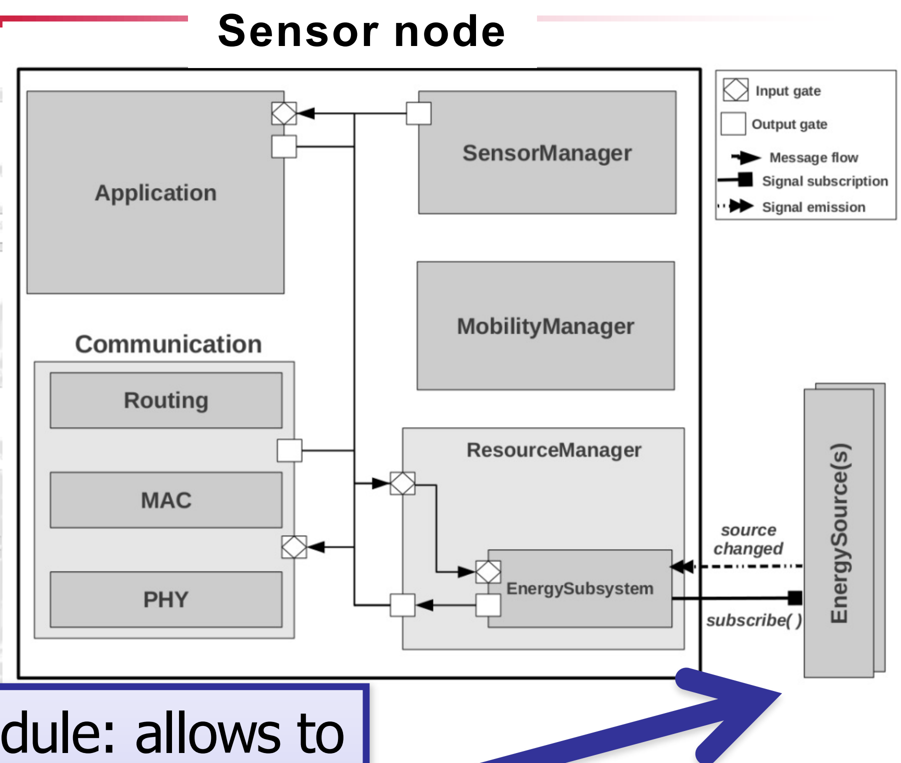


Timer-based contention:

- random jitter computed based on harvesting rate, energy reservoir, and hop count
- Higher priority given to nodes experiencing energy peaks

GreenCastalia: Motivation

- **GreenCastalia features**
  - Support for multi-source harvesting
  - Support for multi-storage devices
  - Support for energy predictions
  - Easily customizable
  - Based on Castalia / OMNET++

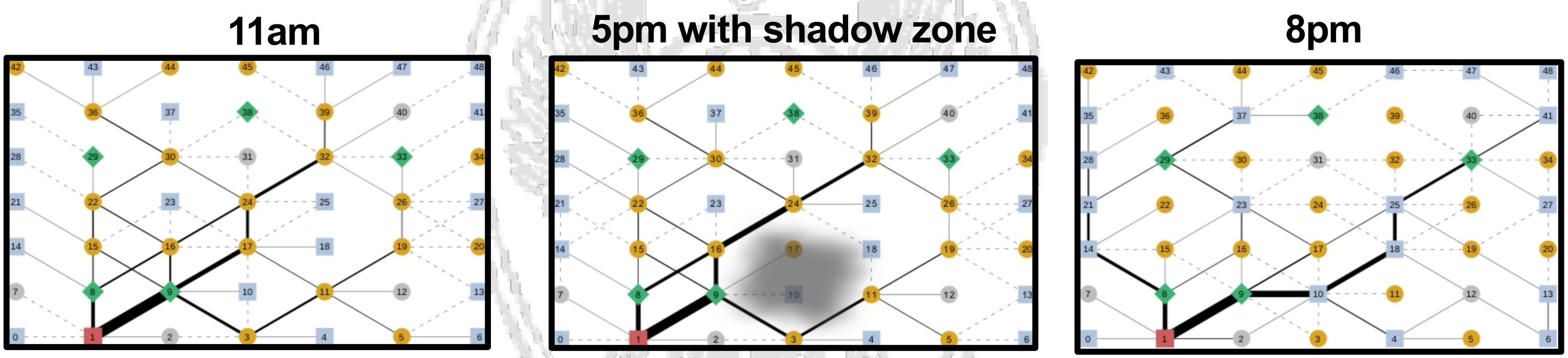


TraceEnergySource module: allows to feed the simulator with timestamped power traces collected through real-life deployments, or with energy availability traces obtained by data repositories or meteorological stations

Harvesting-aware routing: Results

Simulation settings

- 120x120 meters field (7x7 grid deployment)
- Nodes with heterogeneus energy harvesting capabilities:
  - solar, wind both, none



Self-adaptive behaviour: nodes experiencing energy peaks are selected with higher priority as next hop relays

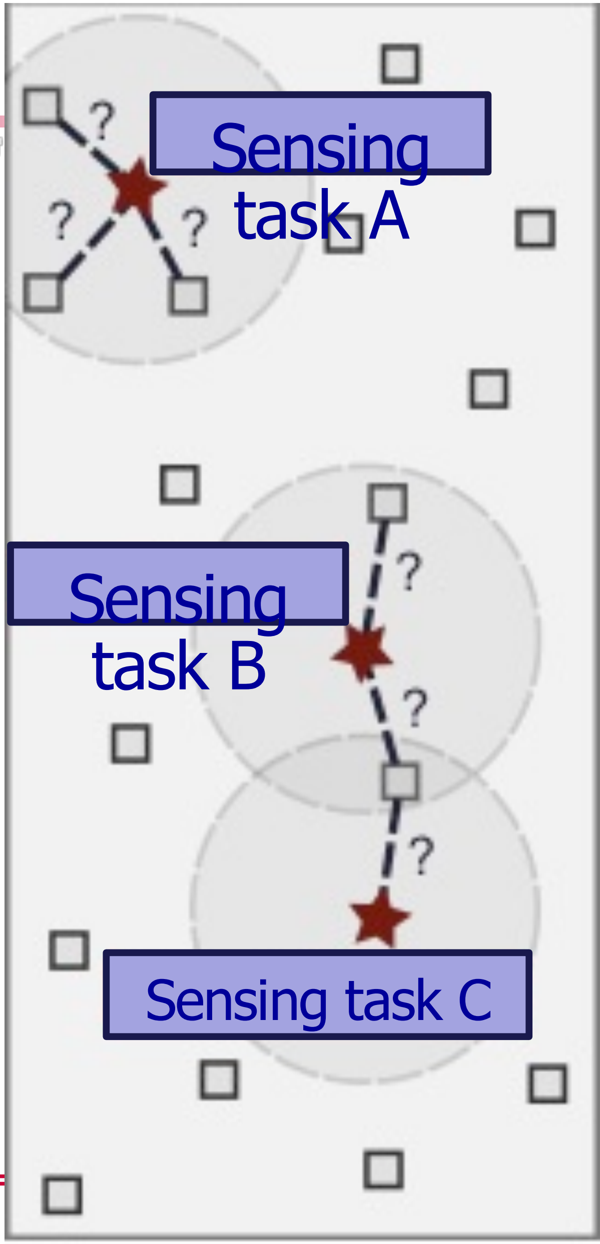
Mission Assignment optimization

Thomas La Porta, Chiara Petrioli, Cinthia Phillips and Dora Spenza, **Sensor-mission assignment in rechargeable wireless sensor networks**. In ***ACM Transactions on Sensor Networks***, Volume 10, Issue 4, Article 60, June 2014.

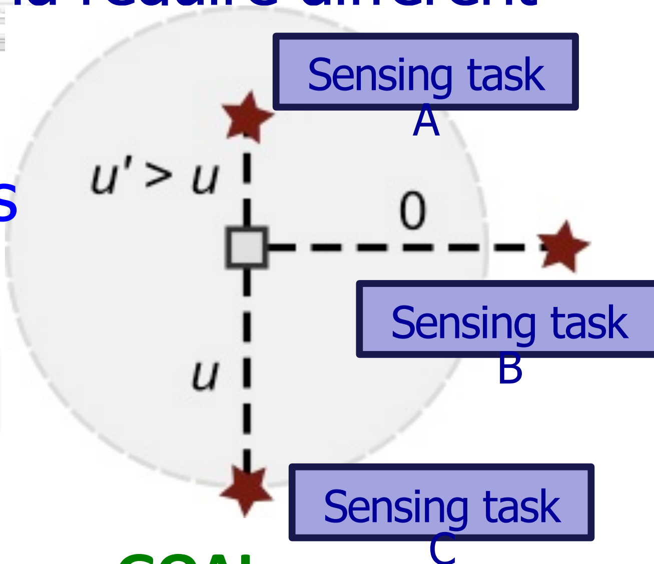
Task allocation

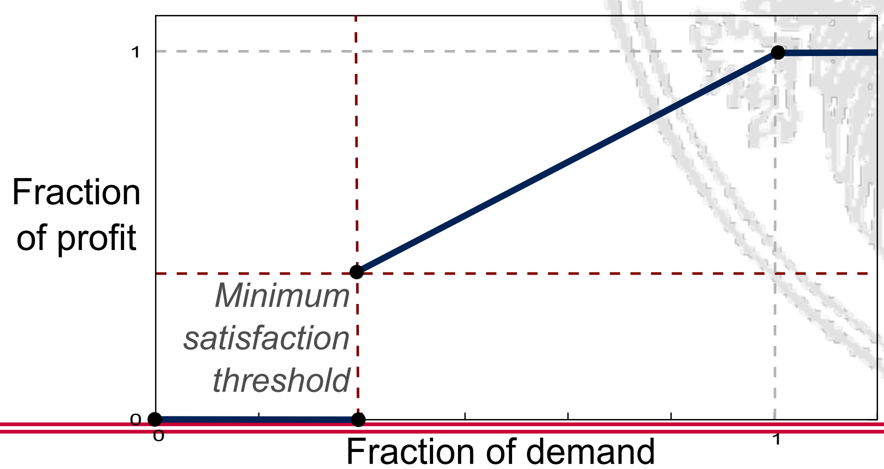
- Sensing tasks (missions) arrive in the network dynamically over time at different locations
- Multiple missions active at the same time, competing for the sensing resources of the network

Decide which sensor(s) should be assigned to each mission



- Missions have different priority (profit) and require different amount of resources (demand)
- **Assigments are not all equal..**
  - Nodes contribute to different missions with different **utility (quality of information)**
  - Achieved profit depends on allocated demand





GOAL

Maximize the profit obtained by the network for missions execution within a given **target lifetime**

EN-MASSE

- Distributed heuristic for task allocation in WSN with **energy harvesting**
- Nodes make independent decisions about task execution
- Decision based on:

Partial profit

1. Profit of the mission
2. Potential contribution to the mission

Tune eagerness

1. Target network lifetime

Classify missions

1. Current energy level of the node (fuel cell + supercap)
2. Energetic cost of the mission
3. Future energy availability

Mission classification

A new mission arrives → check energy requirements and energy availability

More willing to accept ↓

- **Fuel cell/battery required** not enough energy in the supercapacitor to execute the mission; use energy from the fuel-cell
- **Capacitor sustainable** mission energy cost sustained by supercapacitor
- **Recoverable** mission energy cost sustained by supercapacitor AND energy cost recovered through harvesting before the next mission arrives
- **Free** mission energy cost expected to be fully sustained by energy harvesting

Mission classification

A new mission arrives → check energy requirements and energy availability

More willing to accept

- **Fuel cell/battery required** not enough energy in the supercapacitor to execute the mission; use energy from the fuel-cell
- **Capacitor sustainable** mission energy cost sustained by supercapacitor
- **Recoverable** mission energy cost sustained by supercapacitor AND energy cost recovered through harvesting before the next mission arrives
- **Free** mission energy cost expected to be fully sustained by energy harvesting

REQUIRE ENERGY PREDICTIONS

Mission selection rule

capacitor sustainable and recoverable

- Expected partial profit of a mission

$$\overline{p} = \frac{E[u]}{E[d]} \times \frac{E[p]}{P},$$

Always for free missions

P maximum achievable profit: E[u],E[d],E[p] expected utility, demand and profit of a given mission

- Partial profit achievable by a node participating to a mission

$$p^* = \frac{e_{ij}}{d_j} \times \frac{p_j}{P} \times w$$

w weight which depends on mission classification. Bid if p*>=expected partial profit

Task-Allocation EN-MASSE-In summary

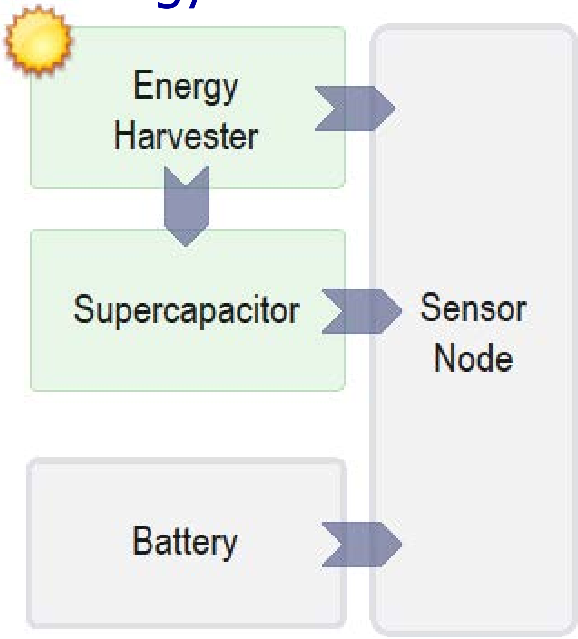
A **decentralized** harvesting-aware heuristic

Key features:

- Uses short and long term energy predictions for pro-active energy allocation
- Takes into account missions arrival statistics to make sustainable allocation decisions
- Considers the impact of executing a mission on node energy

Higher priority to less-impacting missions

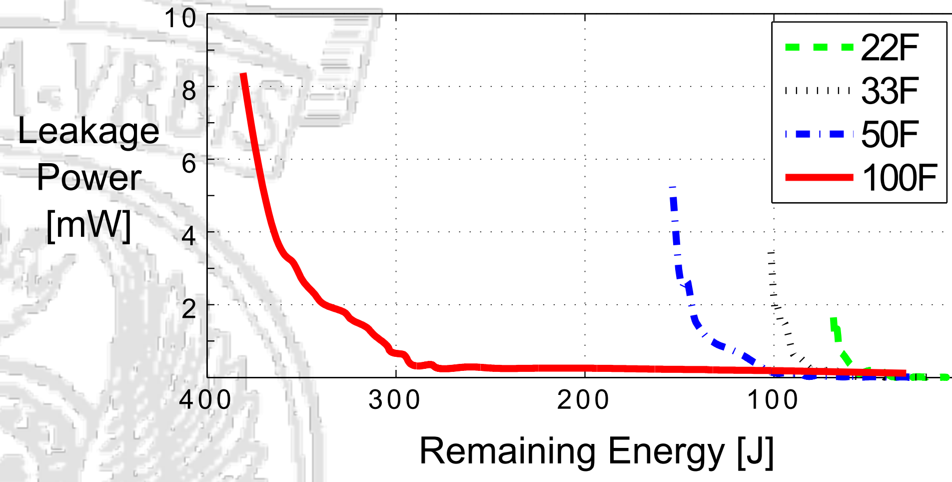
1. **Free:** fully sustained by harvesting
2. **Recoverable:** sustained by supercapacitor and recovered before next mission
3. **Capacitor-sustainable:** sustained by supercapacitor
4. **Battery-required:** sustained by battery



Modeling real harvesting systems

Non-ideal supercapacitors

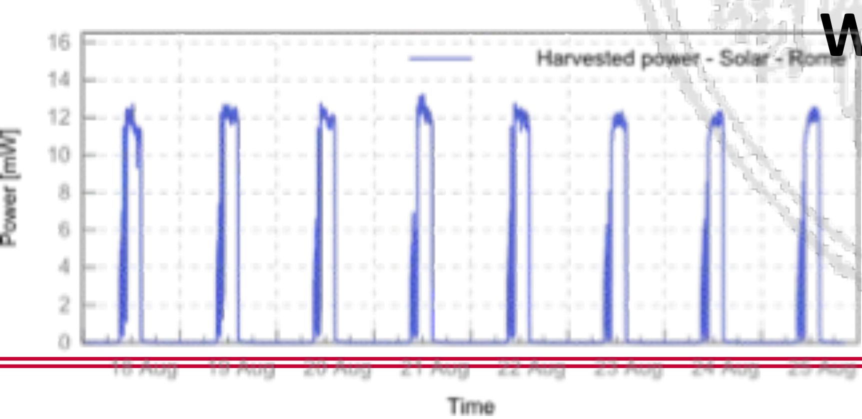
1. Finite size
2. Charging\discharging efficiency < 1
3. Leakage\self-discharge

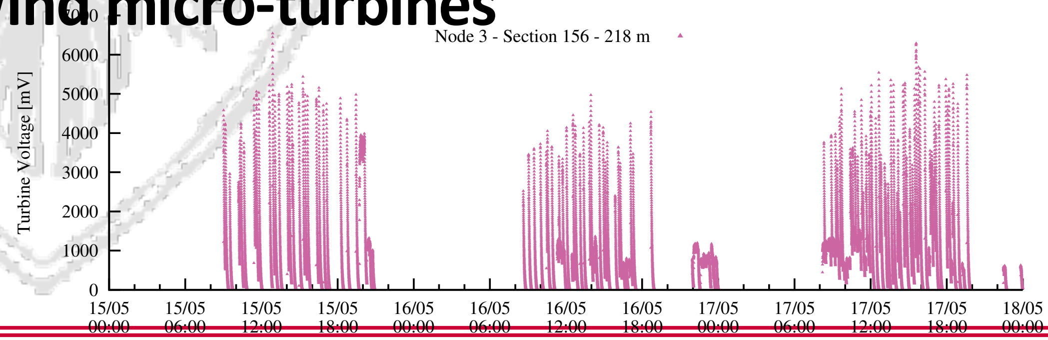


Real-life energy traces

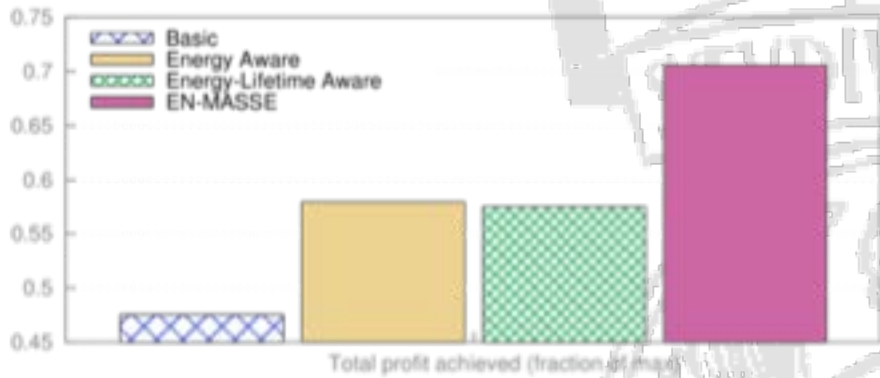
Photovoltaic cells

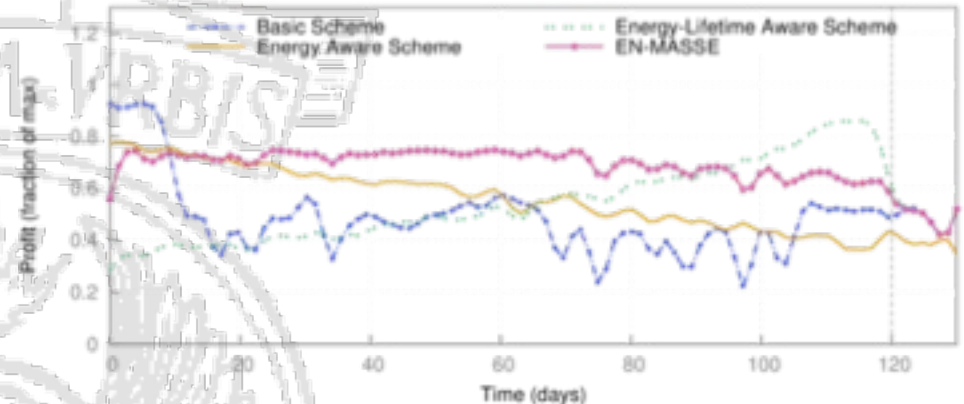
Wind micro-turbines





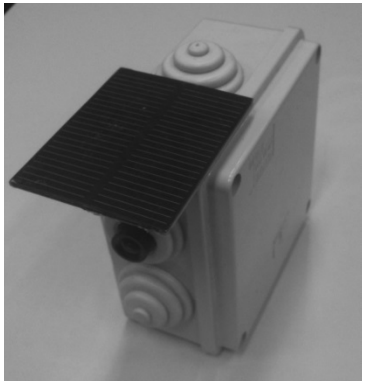
Performance evaluation

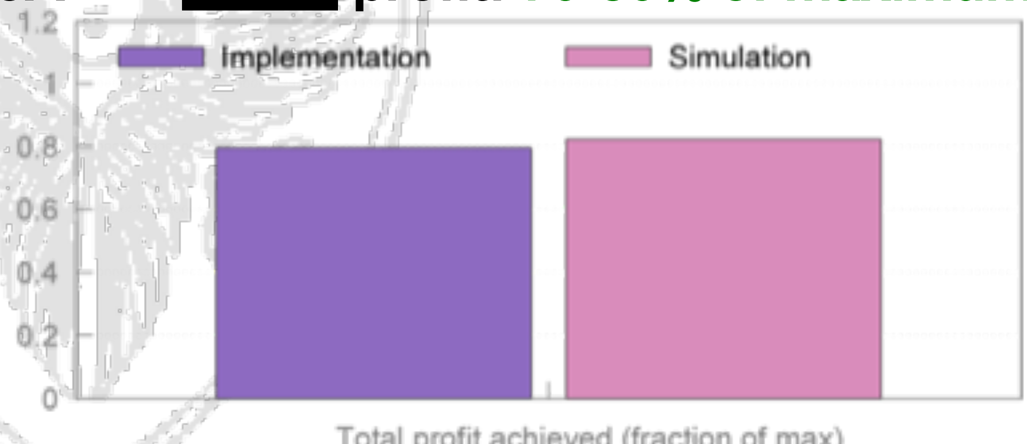




Profit: up to 60% higher than SoA

<u>Stable</u> profit: 70-80% of maximum





In-field testbed validation

Gap between simulations and testbed: **less than 3%** of maximum profit

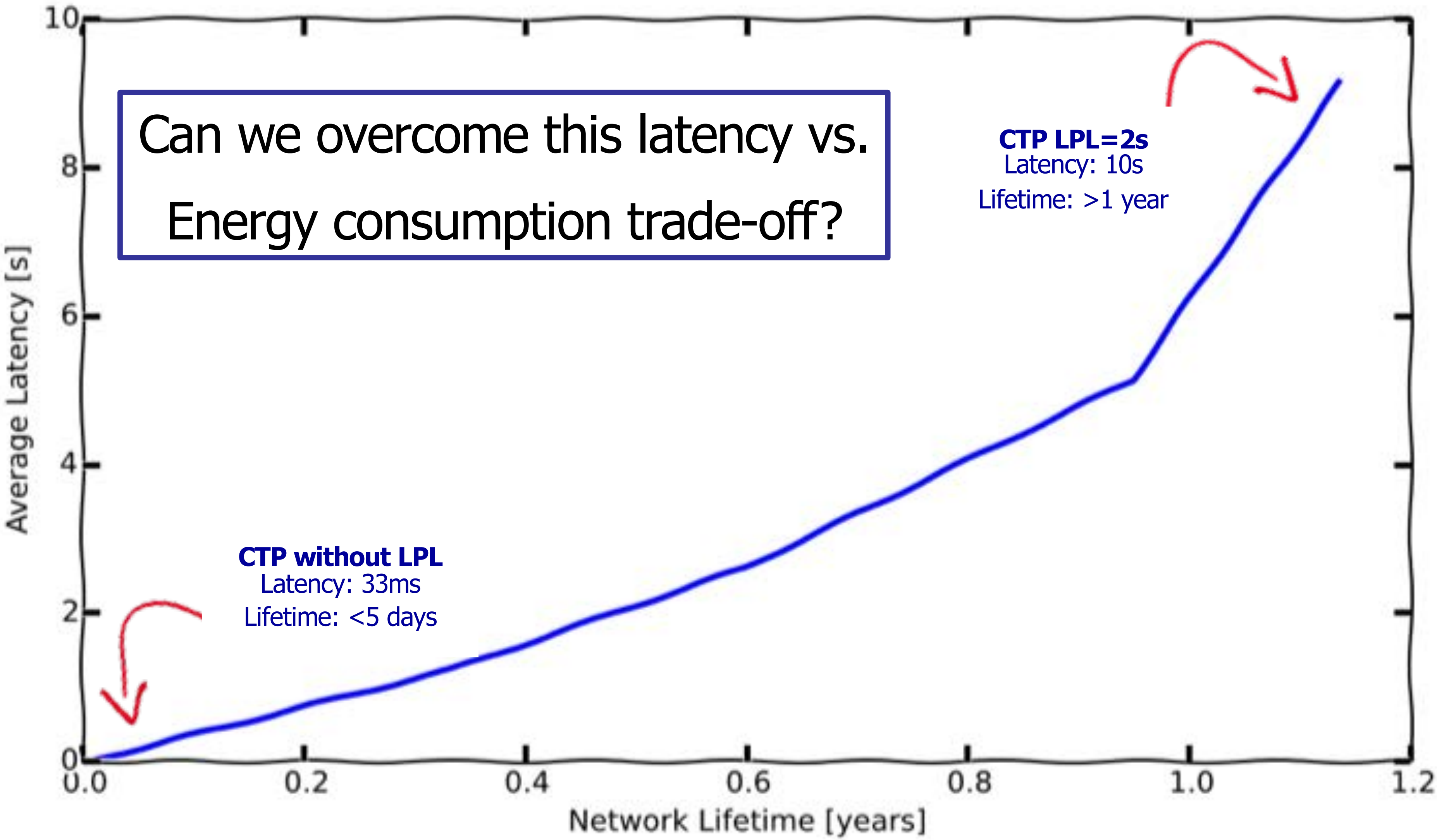
Protocols for wake-up radio enabled Internet of Things, a.a. 2018/2019

Un. of Rome "La Sapienza"

Chiara Petrioli[†]

[†] *Department of Computer Science – University of Rome "Sapienza" – Italy*

Latency vs. Energy Trade-off

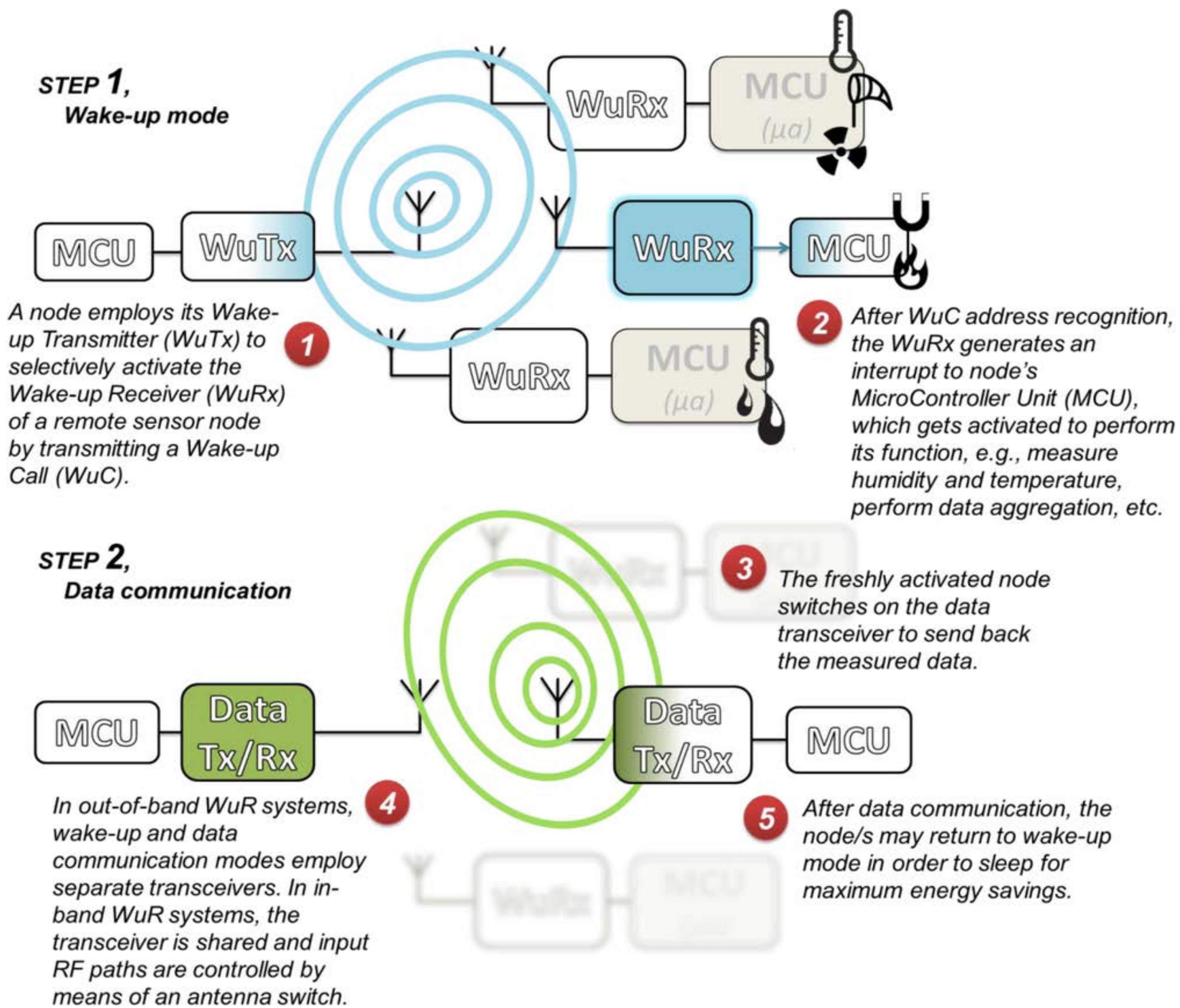


Can we overcome this latency vs. Energy consumption trade-off?

A new approach: Wake-up radios

- Enable **on-demand** communication
  - Low-power dedicated hardware, continuously monitoring the channel
  - Nodes keep their main radio OFF unless data communication is needed
  - Virtually eliminates idle listening on the main radio
  - Based on the architecture, possibility to selectively wake-up only specific nodes
- Terrific energy saving especially in event-based applications
- No latency vs. energy trade off

WRx-based communication



State of the art

Passive WURs

- Harvest power from the radio signal
- No external power supply
- Low sensitivity = short wake-up range (3 m)
- Prone to interferences

RFID-based

- Shifts energy toll to the transmitter
- Unsuitable for P2P networking

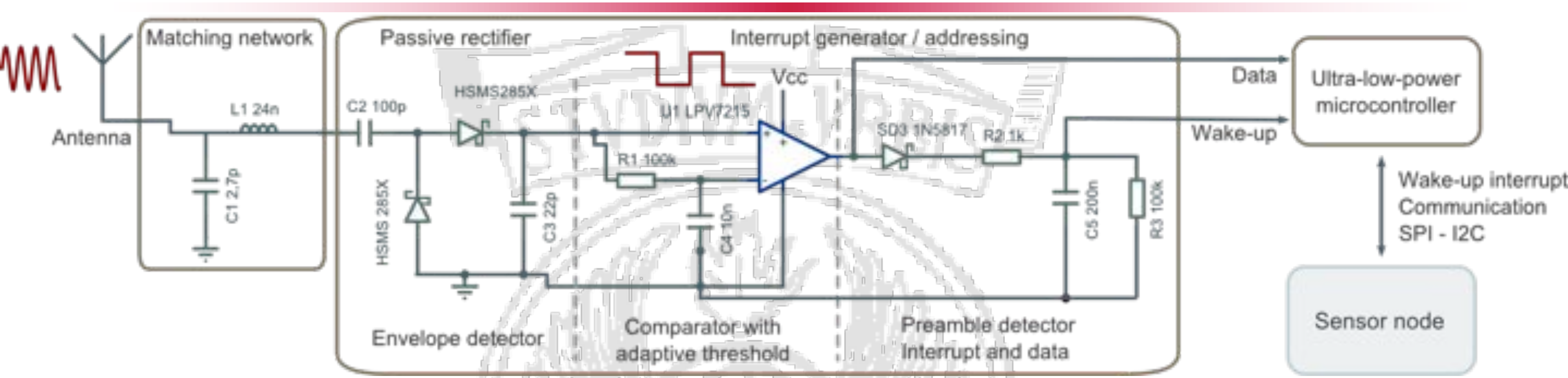
Semi-active WURs

- External power is needed
- Higher sensitivity = longer wake-up range
- Sensitivity: -35 to -47 dBm
- Power consumption: 2.3 to 10 uW

Nano-power WURs

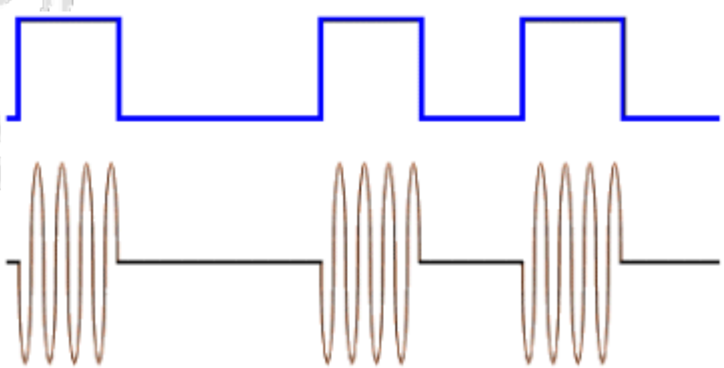
- Power consumption: 98-270 nW
- Wake-up range <= 10m

Our wake-up radio architecture



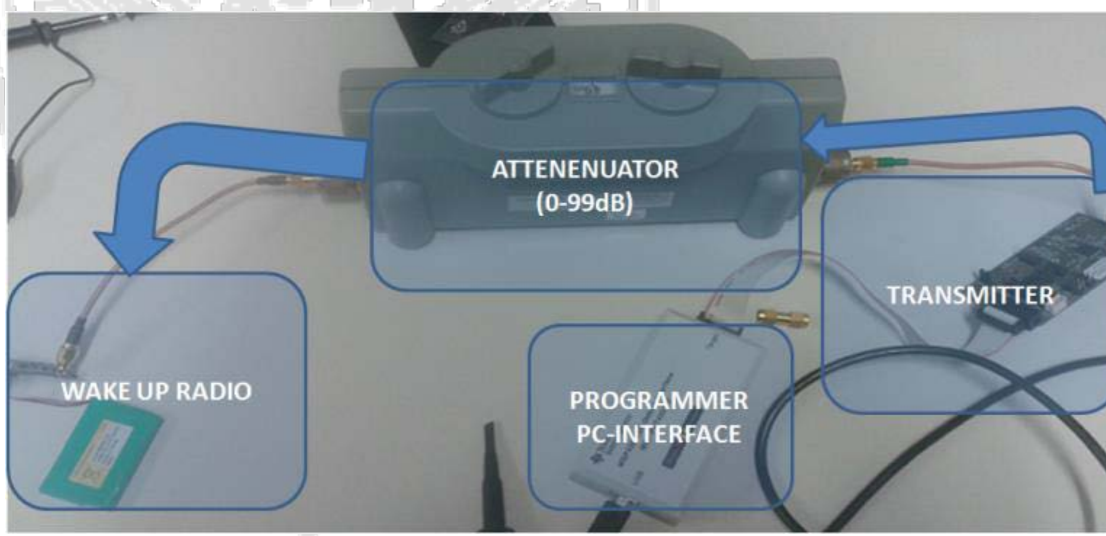
Collaboration with L. Benini and M. Magno, ETHZ

- OOK modulation
- Very low power consumption (< 1.3uW)
- High sensitivity (up to -49dBm)
- Fast reactivity (wake-up time of 130us)
- **Selective addressing**

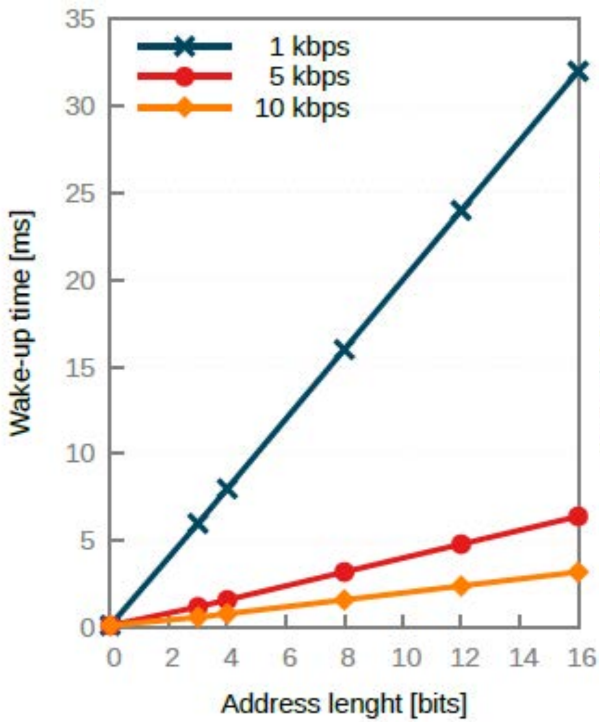


WRx prototyping and testing

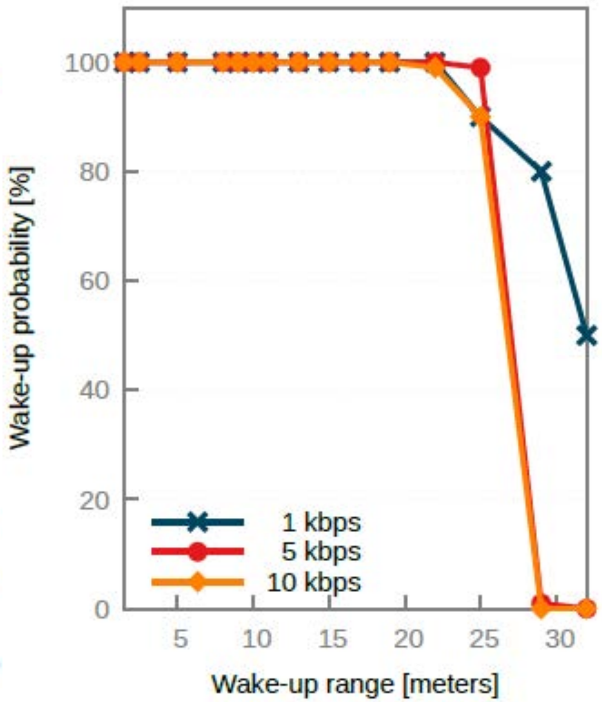
- TI CC1101 used to transmit WRx requests
- on-board PIC microcontroller to perform addressing while keeping the MagoNode in deep-sleep



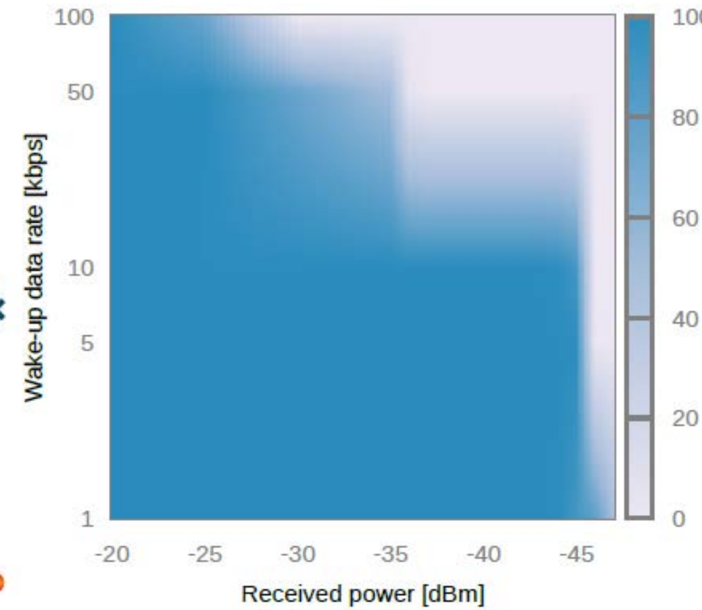
Experimental performance



(a) Wake-up latency



(b) Wake-up probability vs. distance



(c) Wake-up probability vs. received power

- Sensitivity: -49 dBm
- Maximum wake-up range: 42 m (no addressing)
- Wake-up probability depends on WTx data rate and distance

SAPIENZA
UNIVERSITÀ DI ROMA

Chiara Petrioli, Dora Spenza, Pasquale Tommasino and Alessandro Trifiletti, **A Novel wake-up Receiver with Addressing Capability for Wireless Sensor Nodes**. In *Proceedings of the 10th IEEE International Conference on Distributed Computing in Sensor Systems*, ***IEEE DCOSS 2014***, Marina del Rey, USA, May 26-28, 2014, pp. 18-25.

WRx communication protocol

- **Key idea: Semantic wake up addressing**
- use WRx addresses to wake up a node or a group of nodes based on their state
  - selectively wake up only good potential relays
  - e.g., relays ranked based on advance toward the sink, traffic and channel conditions
  - WRx addresses have a semantic meaning
  - each node dynamically changes its own WRx address to reflect its state

Wake up enabled comm. stack

Wake-up-enabled communication stack

- Exploits proposed WuR to addresses latency vs. energy consumption tradeoff
- Both interest dissemination and convergecasting primitives
- Key idea: use wake-up addresses to wake up a node or a group of nodes based on certain properties

Interest dissemination

- Transmission of commands from sink to nodes
- Goal: avoid reception of duplicated packets
- Use current wake-up address to indicate whether a packet was already received

FLOOD-WUP

- Nodes are assigned shared wake-up broadcast addresses: w_a and w_b
- Initially in sleep, wake-up radio active with address $= w_a$
- Sink broadcasts first interest packet preceding it with wake-up sequence w_a
- Nodes with address w_a wakes up, sets main radio to RX, receive packet
- Then change broadcast wake-up address to w_b
- After a random time, nodes re-broadcast packet preceding it with w_a
- No duplicates, only nodes with address w_a wake up

Flood-WUP

FLOOD-WUP

- Nodes are assigned shared wake-up broadcast addresses: w_a and w_b
- Initially in sleep, wake-up radio active with address $= w_a$
- Sink broadcasts first interest packet preceding it with wake-up sequence w_a
- Nodes with address w_a wakes up, sets main radio to RX, receive packet
- Then change broadcast wake-up address to w_b
- After a random time, nodes re-broadcast packet preceding it with w_a
- No duplicates, only nodes with address w_a wake up

FLOOD-WUP

- Nodes are assigned shared wake-up broadcast addresses: w_a and w_b
- Initially in sleep, wake-up radio active with address = w_a
- Sink broadcasts first interest packet preceding it with wake-up sequence w_a
- Nodes with address w_a wakes up, sets main radio to RX, receive packet
- Then change broadcast wake-up address to w_b
- After a random time, nodes re-broadcast packet preceding it with w_a
- No duplicates, only nodes with address w_a wake up

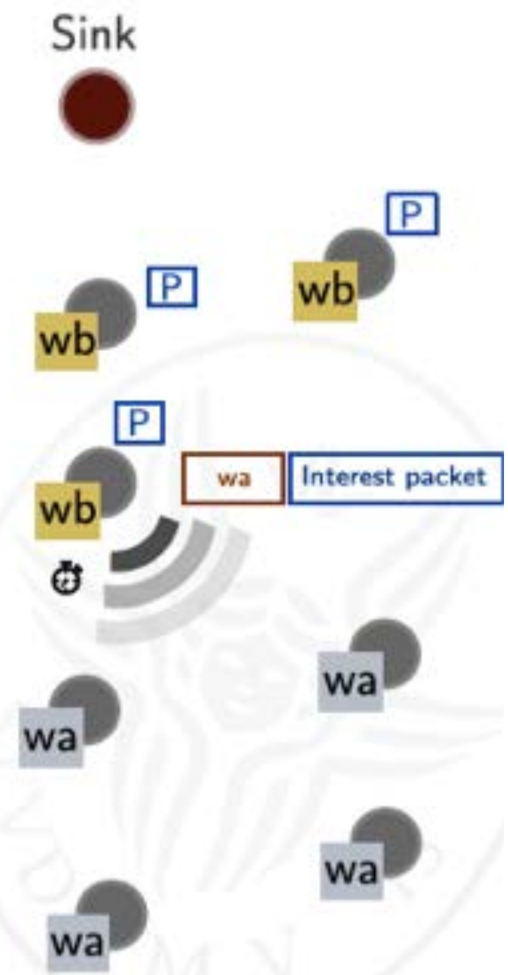
Flood-WUP

FLOOD-WUP

- Nodes are assigned shared wake-up broadcast addresses: w_a and w_b
- Initially in sleep, wake-up radio active with address = w_a
- Sink broadcasts first interest packet preceding it with wake-up sequence w_a
- Nodes with address w_a wakes up, sets main radio to RX, receive packet
- Then change broadcast wake-up address to w_b
- After a random time, nodes re-broadcast packet preceding it with w_a
- No duplicates, only nodes with address w_a wake up

FLOOD-WUP

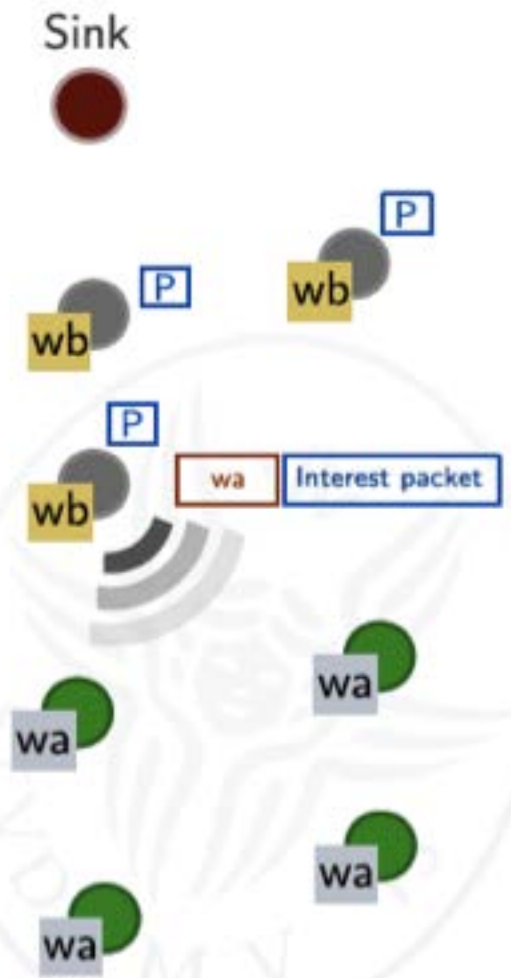
- Nodes are assigned shared wake-up broadcast addresses: w_a and w_b
- Initially in sleep, wake-up radio active with address $= w_a$
- Sink broadcasts first interest packet preceding it with wake-up sequence w_a
- Nodes with address w_a wakes up, sets main radio to RX, receive packet
- Then change broadcast wake-up address to w_b
- After a random time, nodes re-broadcast packet preceding it with w_a
- No duplicates, only nodes with address w_a wake up



Flood-WUP

FLOOD-WUP

- Nodes are assigned shared wake-up broadcast addresses: w_a and w_b
- Initially in sleep, wake-up radio active with address $= w_a$
- Sink broadcasts first interest packet preceding it with wake-up sequence w_a
- Nodes with address w_a wakes up, sets main radio to RX, receive packet
- Then change broadcast wake-up address to w_b
- After a random time, nodes re-broadcast packet preceding it with w_a
- No duplicates, only nodes with address w_a wake up



Converge Casting: GREEN-WUP

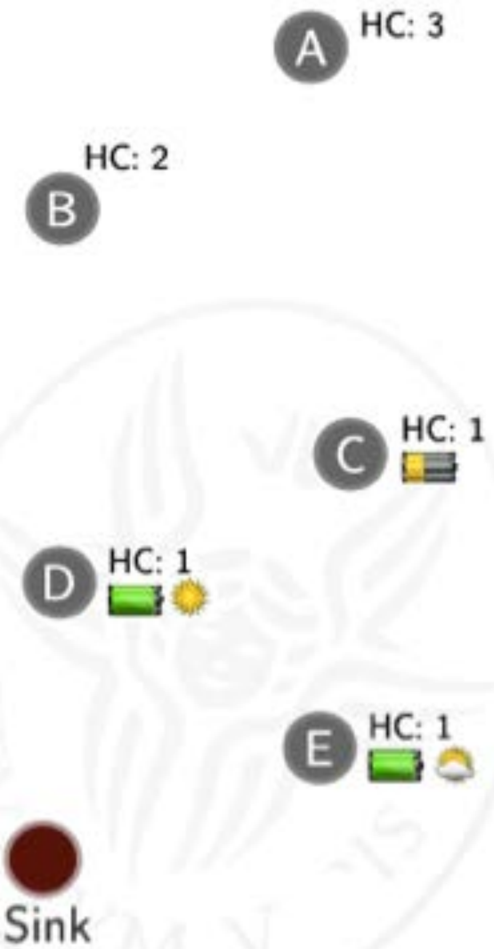
- Multi-hop data transfer to sink
- Energy harvesting scenario: nodes scavenge power from environment
- Key idea: Selectively wake-up only good potential relays
- Ranked based on hop count, residual energy, energy intake (harvesting-aware)
- Wake-up addresses have a semantic meaning
- Nodes dynamically change their wake-up addresses over time to reflect their state
  - Format | Hop count | Energy class |
  - Energy class depends on harvesting and residual energy

GREEN-WUP

- Example: energy classes

  max battery level > threshold T_{high} and excess energy from harvesting
  2 battery level > T_{high}
  1 T_{low} < battery level $\leq T_{high}$
  0 battery level $\leq T_{low}$

- Node B has a packet to transmit
- Sends RTS only to nodes with hopcount = 1 and energy = max. Then it goes to sleep
- Only node D wakes up. Other nodes continues to sleep
- Iterate on energy class if no relay found
- CTS, DATA, ACK...

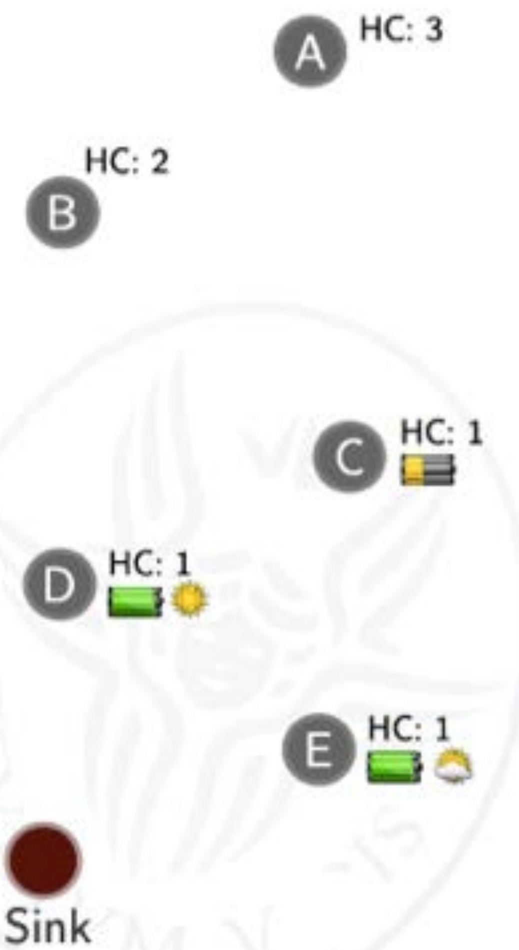


GREEN-WUP

- Example: energy classes

  max battery level $>$ threshold T_{high} and excess energy from harvesting
  2 battery level $> T_{high}$
  1 $T_{low} <$ battery level $\leq T_{high}$
  0 battery level $\leq T_{low}$

- Node B has a packet to transmit
- Sends RTS only to nodes with hopcount = 1 and energy = max. Then it goes to sleep
- Only node D wakes up. Other nodes continues to sleep
- Iterate on energy class if no relay found
- CTS, DATA, ACK...

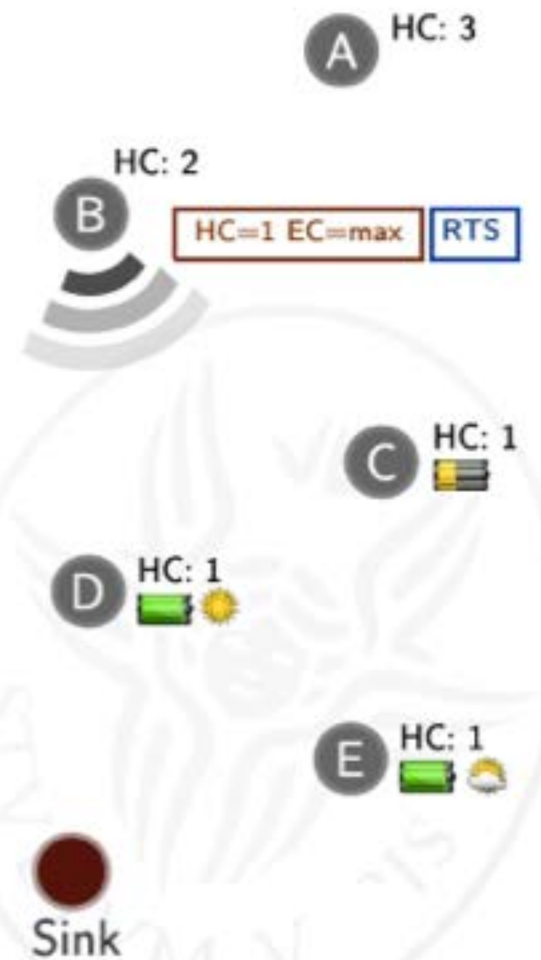


GREEN-WUP

- Example: energy classes

  max battery level > threshold T_{high} and excess energy from harvesting
  2 battery level > T_{high}
  1 T_{low} < battery level $\leq T_{high}$
  0 battery level $\leq T_{low}$

- Node B has a packet to transmit
- Sends RTS only to nodes with hopcount = 1 and energy = max. Then it goes to sleep
- Only node D wakes up. Other nodes continues to sleep
- Iterate on energy class if no relay found
- CTS, DATA, ACK

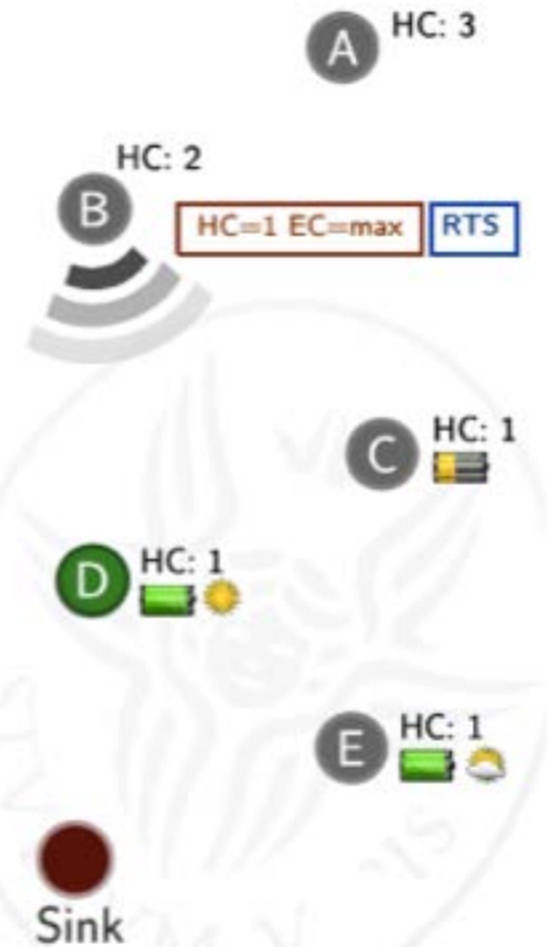


GREEN-WUP

- Example: energy classes

  max battery level > threshold T_{high} and excess energy from harvesting
  2 battery level > T_{high}
  1 T_{low} < battery level ≤ T_{high}
  0 battery level ≤ T_{low}

- Node B has a packet to transmit
- Sends RTS only to nodes with hopcount = 1 and energy = max. Then it goes to sleep
- Only node D wakes up. Other nodes continues to sleep
- Iterate on energy class if no relay found
- CTS, DATA, ACK..

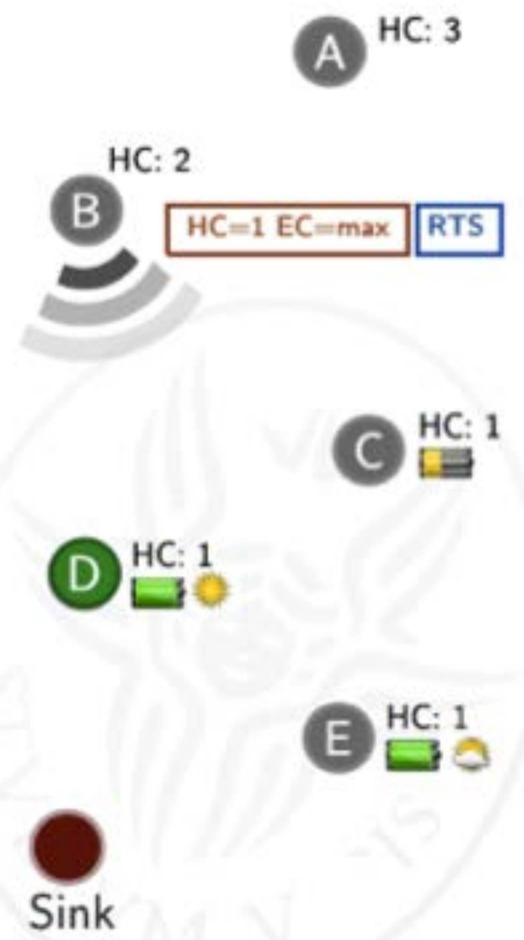


GREEN-WUP

- Example: energy classes

  max battery level > threshold T_{high} and excess energy from harvesting
  2 battery level > T_{high}
  1 T_{low} < battery level ≤ T_{high}
  0 battery level ≤ T_{low}

- Node B has a packet to transmit
- Sends RTS only to nodes with hopcount = 1 and energy = max. Then it goes to sleep
- Only node D wakes up. Other nodes continues to sleep
- Iterate on energy class if no relay found
- CTS, DATA, ACK..

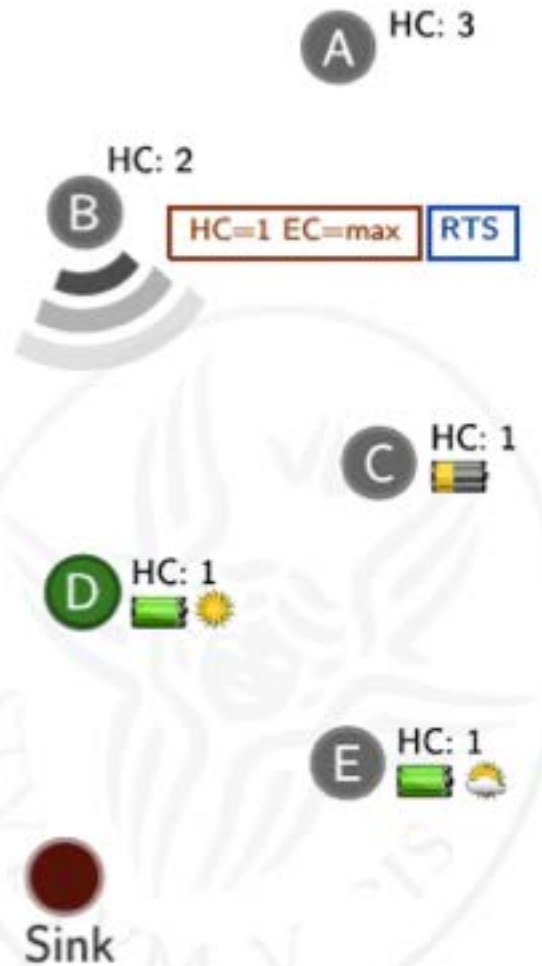


GREEN-WUP

- Example: energy classes

  max battery level > threshold T_{high} and excess energy from harvesting
  2 battery level > T_{high}
  1 T_{low} < battery level ≤ T_{high}
  0 battery level ≤ T_{low}

- Node B has a packet to transmit
- Sends RTS only to nodes with hopcount = 1 and energy = max. Then it goes to sleep
- Only node D wakes up. Other nodes continues to sleep
- Iterate on energy class if no relay found
- CTS, DATA, ACK..



SAPIENZA
UNIVERSITÀ DI ROMA

Sim. scenario

Different concept of wake up radio

- Active, higher energy consumption

+ Semantic WUP radio addresses

+ Only one transceiver (TX)

Key feature of the proposed WuR

- Selective addressing of nodes combining frequency-domain and time-domain addressing space
- **Wake-up signal**: sequence of continuous-wave pulses OOK modulated over c IEEE 802.15.4 channels (2.4 GHz ISM band)

Prototyping and design validation

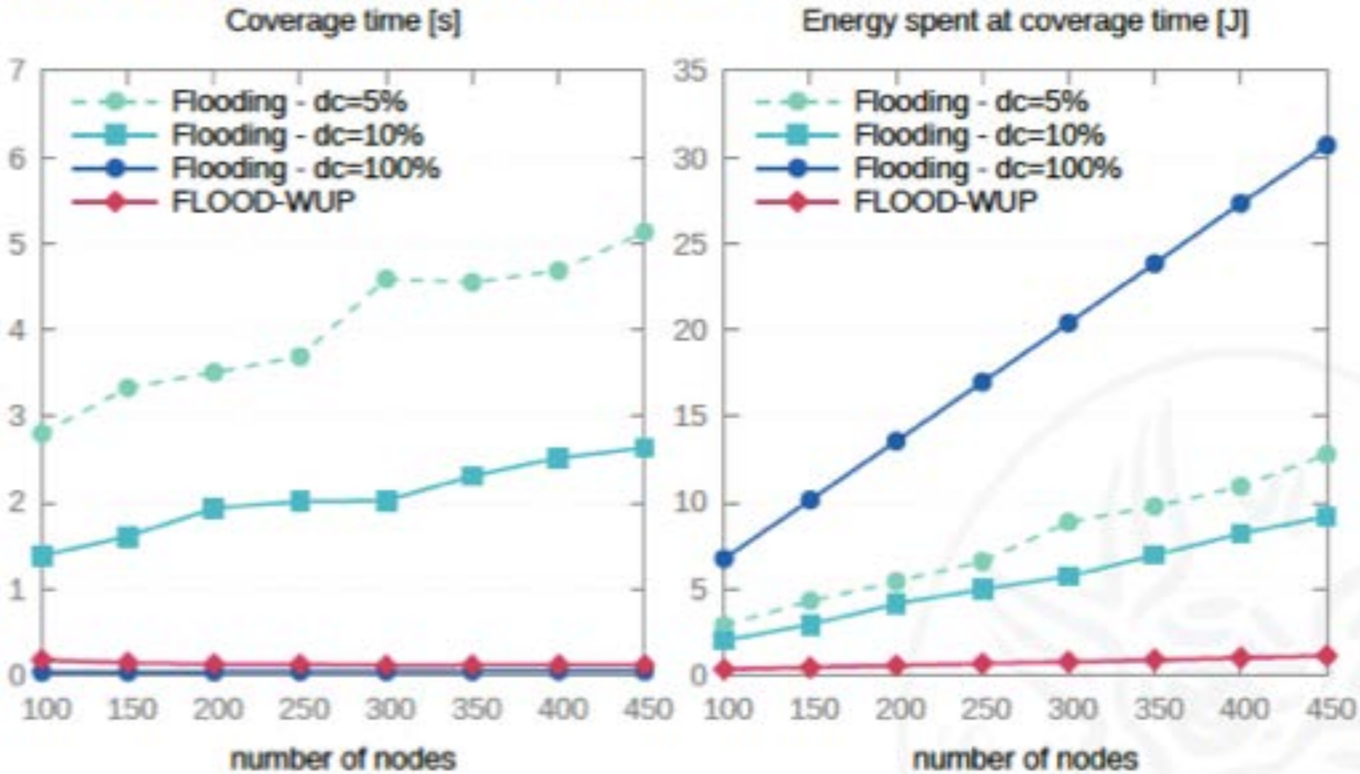
- Prototype with 4 channels at 2410, 2435, 2455 and 2480 MHz
- Lab experiments: sensitivity of -83 dBm
- In-field experiments: RX node (TelosB + WuR), TX node (MTM-CM3300)
- Varying distance up to 120 m
- TX node sends 4 different wake-up sequences (8 symbols) 100 times
- Wake-up statistics recorded by RX node
- False positive and false negative both < 1%
- Simulation for power consumption scaling: 168 μW per filter, 1.6 mW overall (including LNA)

Green Castalia simulations

Chiara Petrioli, Dora Spenza, Pasquale Tommasino, Alessandro Trifiletti

A Novel Wake-Up Receiver with Addressing Capability for Wireless Sensor Nodes. IEEE DCOSS 2014: 18-25

FLOOD-WUP, 100-450 nodes



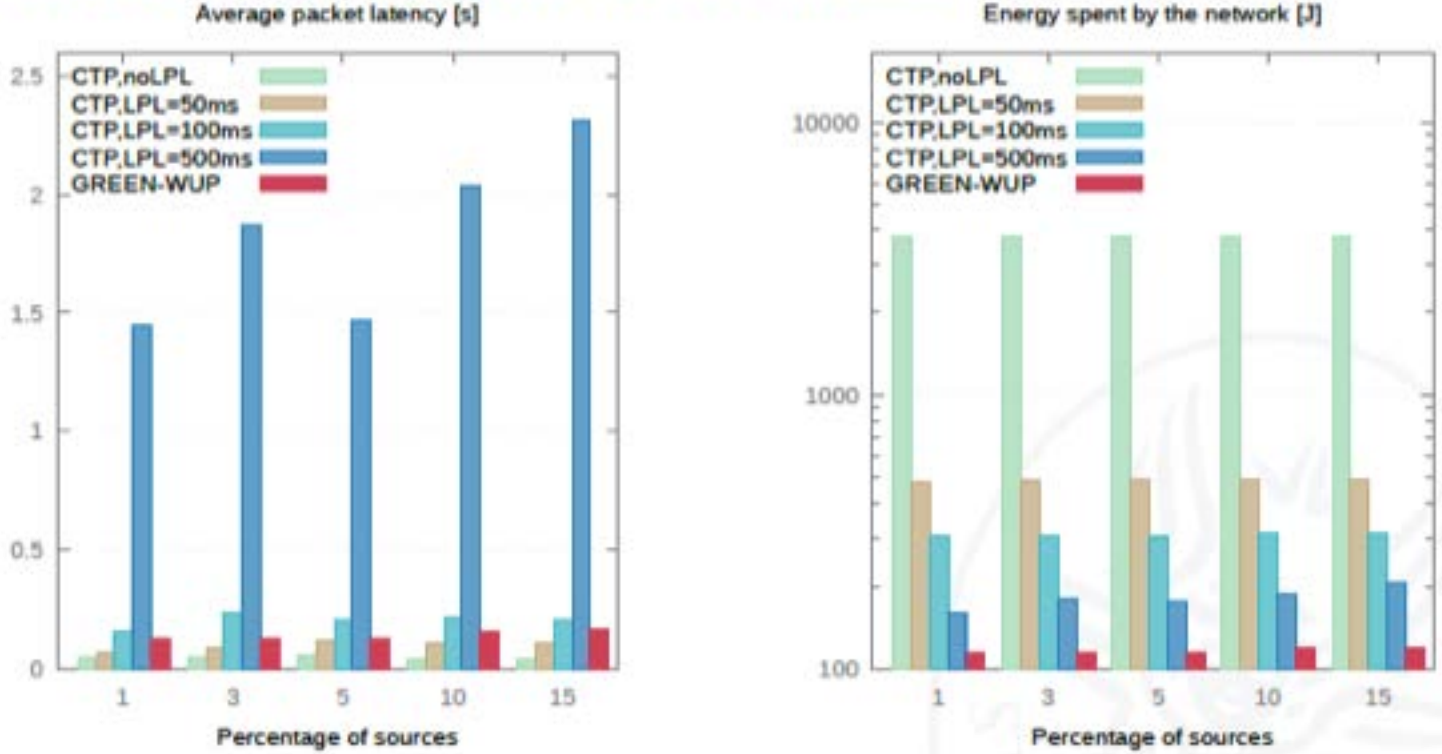
Coverage time similar to Flooding with 100% DC

Energy consumption reduced of up to ≈ 96%

vs Flooding10%: -24x coverage time, -8x energy

Perf. Evaluation

GREEN-WUP, network of 100 nodes



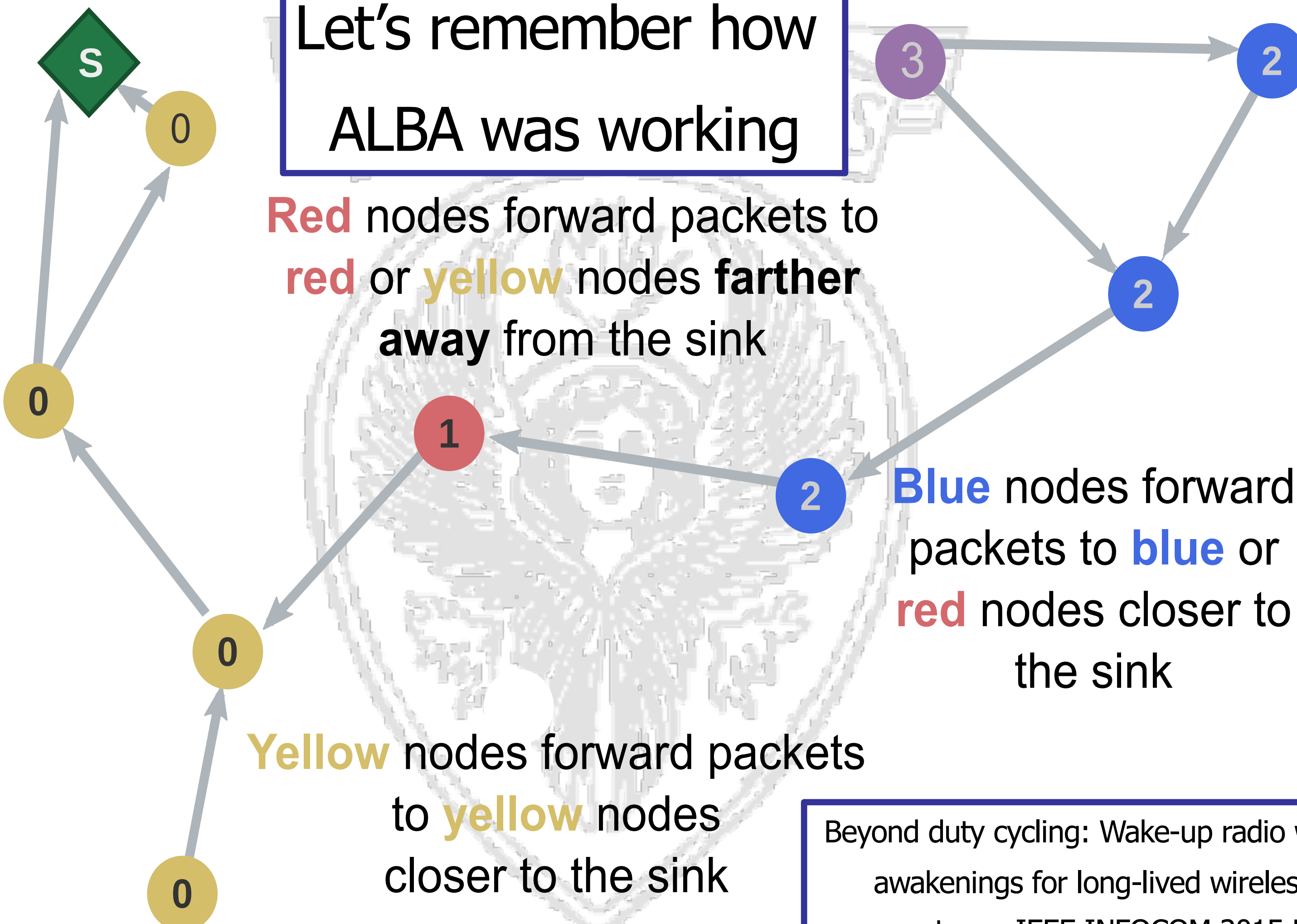
Average latency 100 ms higher than CTP without LPL

Energy consumption reduced of up to $\approx 33x$!

vs CTP+LPL=500ms: latency -16x, -45% energy

Dora Spenza, Michele Magno, Stefano Basagni, Luca Benini, Mario Paoli and Chiara Petrioli, **Beyond Duty Cycling: Wake-up Radio with Selective Awakenings for Long-lived Wireless Sensing Systems**. In Proceedings of ***IEEE INFOCOM 2015***, Hong Kong, 26 April - 1 May, 2015, pp. 522 - 530.

Let's remember how ALBA was working



Red nodes forward packets to **red** or **yellow** nodes **farther away** from the sink

Blue nodes forward packets to **blue** or **red** nodes closer to the sink

Yellow nodes forward packets to **yellow** nodes closer to the sink

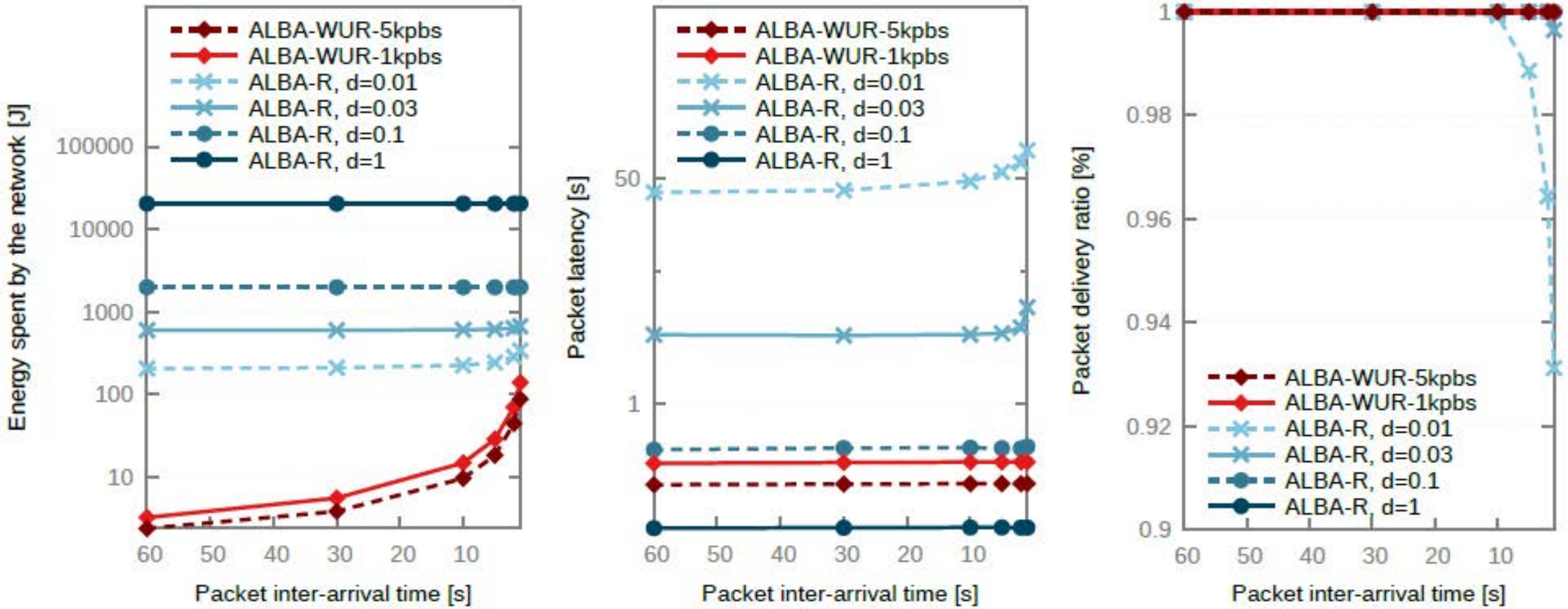
Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems. IEEE INFOCOM 2015 522-530

Simulation setup

- Comparative performance evaluation:
  a. ALBA-WUR, WRx sequences sent @ 1 kbps, 5 kpbs
  b. ALBA-R with duty d={1, 0.1, 0.03, 0.01}
- Simulation framework: GreenCastalia
- 120 nodes distributed randomly and uniformly over a 200x200m field
- WRx modeling based on experimental data
- Nodes powered by 2xAA alkaline batteries with capacity = 2500mAh

SAPIENZA
UNIVERSITÀ DI ROMA

Performance evaluation results



(a) Energy consumption (the y axis is logscale) (b) Average packet latency (the y axis is logscale) (c) Packet delivery ratio

- Energy consumption: reduced up to 5 orders of magnitude
- Latency: always better than ALBA-R with duty cycle < 100%
- PDR: 100%

Expected network lifetime

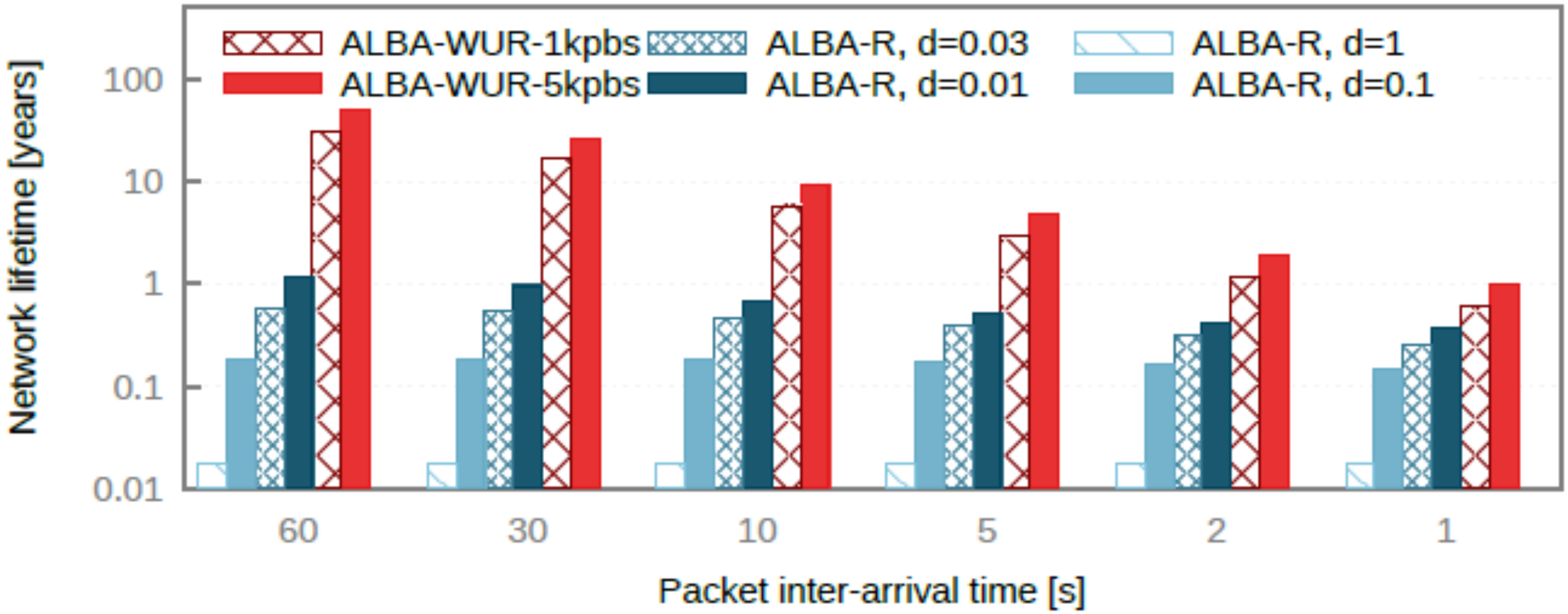


Fig. 5. Network lifetime (the y axis is logscale).

- Lifetime of **several decades**!
- Network with 1% duty cycle and **<u>no traffic</u>** = less than 2 years

What's next….

In the standards…
IEEE 802.11ba group on Wake Up radios

This amendment defines a physical (PHY) layer specification and defines modifications to the medium access control (MAC) layer specification that enables operation of a wake-up radio (WUR). The wake-up frames carry only control information. The reception of the wake-up frame by the WUR can trigger a transition of the primary connectivity radio out of sleep. The WUR is a companion radio to the primary connectivity radio and meets the same range requirement as the primary connectivity radio. The WUR devices coexist with legacy IEEE 802.11 devices in the same band. The WUR has an expected active receiver power consumption of less than one milliwatt.